

<http://WWW.CABARE.NET> ©

Le langage H.T.M.L

- cours -

Introduction au langage HTML

Michel Cabaré – Ver 4.1 – Nov 2009-

TABLE DES MATIERES

OUTILS NECÉSSAIRES	6
Passer de l'éditeur au navigateur	6
EDITEUR TEXTE	7
Wordpad	7
Fichier Nouveau - Ouvrir - Enregistrer	8
NAVIGATEURS	9
Firefox	9
Internet Explorer.....	9
1° PAGE HTML ET TAGS	10
Construction.....	10
Explication des « tags » HTML.....	11
HTML ou XHTML	12
Le squelette d'une page XHTML - HTML.....	13
LES CARACTÈRES SPÉCIAUX	14
Le français et les caractères accentués é-è-ê	14
Table des caractères.....	15
Les caractères spéciaux	15
Codage Unicode	16
Balise Méta charset =iso-xxx	17
Les espaces et les fins de ligne	18
MISE EN FORME DE TEXTE 1°	20
Saut de ligne - taille - gras - italique	20
RETRAITS ET LISTES	22
Retraits	22
Listes non numérotées	24
Listes numérotées	24
INSÉRER DES LIGNES	26
MISE EN FORME DE TEXTE 2°	28
Taille de texte.....	28
Couleur	28
Et encore	30
LES TABLEAUX	31
Construction.....	31
Présentation	32
et encore.....	34
Mise en forme	34
POUR UN FICHIER HTML RELISIBLE.....	36
Commentaires	36
Disposition.....	36



LIEN HYPERTEXTE.....	37
objectifs	37
Adresse mail	38
Liens entres pages HTML.....	39
Emplacement des pages HTML : relatif ou absolu ?.....	42
Découpage de pages HTML longues	43
Paramétrage de l'emplacement	44
Structure linéaire ou hiérarchique	44
IMAGES ET HTML.....	45
Positionner une image directement.....	46
Positionner une image dans un tableau	48
Taille et chargement d'une image	49
image précédée d'une ... image	50
FICHIERS GRAPHIQUES.....	51
Bitmap-Vectoriels	51
Résolution nécessaire	52
Formats GIF, JPEG	52
Formats images principaux.....	53
IMAGES ET LIENS.....	54
Nommer des images.....	55
Combiner des images	56
PARAMETRES DE PAGE.....	57
Couleur - arrière plan	57
Liens hyper-texte	58
FORMULAIRES (CRÉATION).....	59
Structure Générale.....	60
le tag INPUT	61
le tag SELECT	62
le tag TEXTAREA	62
Annulation ou Envoi	63
Présentation	64
FORMULAIRES (TRAITEMENT).....	65
principes	65
formulaire via "mailto"	65
POLICE PROPORTIONELLE OU NON	67
ÉDITEURS & GENERATEURS	68
Editeurs HTML.....	68
Gestionnaires de Site	68
Conversion en HTML.....	68
TEMPS DE ... LECTURE.....	69
Taille des pages	69
Vitesse de chargement	69

COMMENT STRUCTURER UN SITE WEB.....	70
Quoi mettre sur le site Web	70
Comment structurer un site Web	70
Structure linéaire	71
Structure en arbre	71
Structure en réseau	72
Structure mixte	72
La structure des pages Web.....	73
La page d'accueil.....	73
Les pages secondaires	73
LES TAGS HTML (EN LISTE)	74
LES TAGS HTML (PAR FONCTION).....	80
Corps de fichier	80
Mise en forme de texte	81
Listes	82
Lignes horizontales	83
Tableaux	83
Liens	84
Insertion d'Images	84
Formulaires	85
TAGS META (CHARSET).	86
Tags spéciaux pour les caractères nationaux.....	86
TAGS META RÉFÉRENCIEMENT.	88
Revisit After	88
Title	88
Keywords	89
Description	89
Robots	90
Revisit After	90
NORMES HTML.....	91
Principe de tolérance	91
HTML ver 2.0 juin 1994:	91
HTML ver 3.2 Mai 1997:.....	92
HTML ver 4.01 Décembre 99:	92
HTML ou XHTML	92
Le squelette d'une page XHTML - HTML.....	93
Vérification W3c:	93
MULTIMEDIA ET HTML.....	94
Lien simple	94
Plug-in.....	94
Les Balises possibles	95
Balise Embed.....	96
Balise Object	98
type MIME :.....	100
classid et codebase :	103



SON ET HTML	104
Balise Object et Embed	104
VIDEO ET HTML	105
Video AVI.....	105
FLASH ET HTML	106
Incorporation de flash	106
Paramètres flash player.....	106
Vidéo swf dans...flash.....	109
Vidéo .flv et player flash flv	109
FEUILLES DE STYLE - CSS	111
Historique des CSS :	111
Avantages des feuilles de style :.....	112
Pourquoi "En Cascade" :	112
Lien vers une feuille de styles.....	113
Les sélecteurs sur balises HTML	113
principales balises html redéfinissables.....	114
QU'EST-CE JAVASCRIPT	115
Présentation	115
Objectifs.....	115
Les plus et ...les moins :.....	116
QU'EST-CE PHP	117
Ce qu'est php	117
Les plus et ...les moins :.....	117
CARACTÈRES SPÉCIAUX HTML	118
Table des caractères.....	118
CONVERSION DE VIDEO QCM	119
exemple Quick Media Converter.....	119

OUTILS NÉCESSAIRES

Dans un premier temps, pour créer des pages HTML sur le WEB il est suffisant d'avoir deux éléments :

- Un éditeur de texte commun : pour taper les documents
- Un navigateur au choix : pour visualiser ces documents

Sur P.C. sous Windows, on pourra travailler avec WordPad, l'éditeur texte fournit avec le système d'exploitation, étant bien entendu que n'importe quel éditeur pourrait faire l'affaire.

On construira des fichiers que l'on enregistrera classiquement sur le disque.

Comme navigateur on utilisera dans ce support Netscape Navigator, étant bien entendu que n'importe quel navigateur pourrait faire l'affaire

Dans un deuxième temps pour créer des pages HTML sur le WEB il est possible d'utiliser un éditeur HTML permettant de travailler en pré-visualisation immédiate des pages construites et de mieux gérer les liens hyper texte construits dans les pages.

Passer de l'éditeur au navigateur

Une manipulation systématique de l'éditeur et du navigateur étant nécessaire, on lancera les deux applications simultanément, et on passera de l'une à l'autre via la barre des tâches

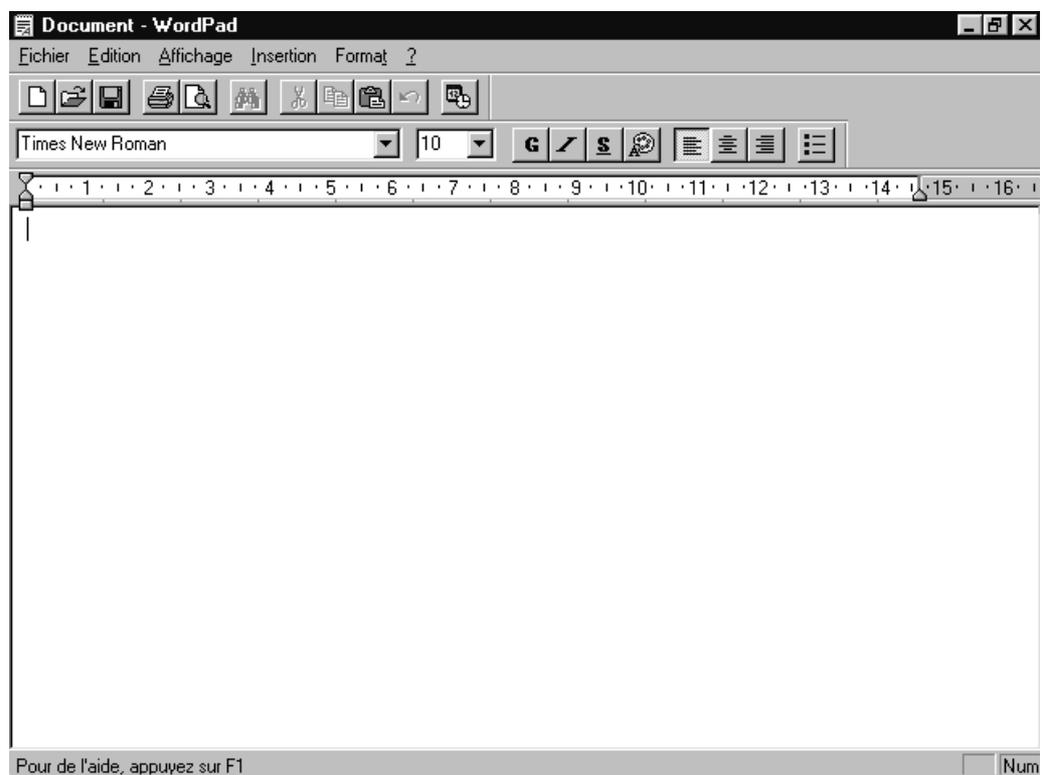


EDITEUR TEXTE

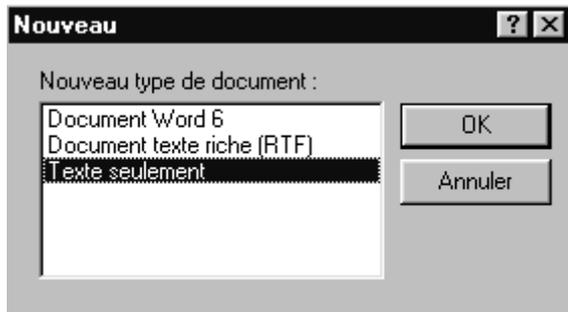
Wordpad

Pour démarrer Wordpad demander le Menu :

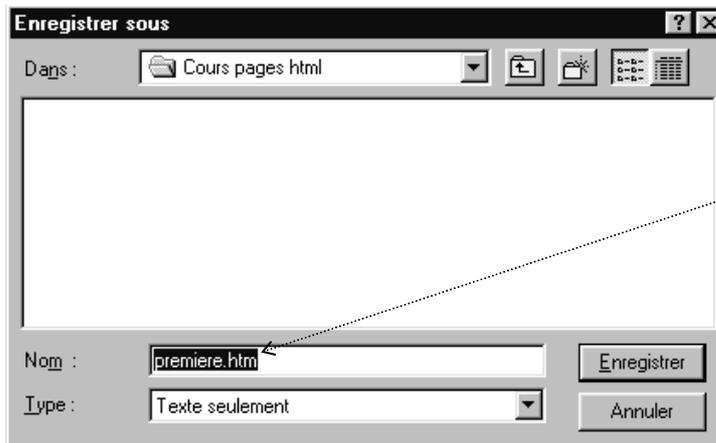
Démarrer / Programme / Accessoires / Wordpad



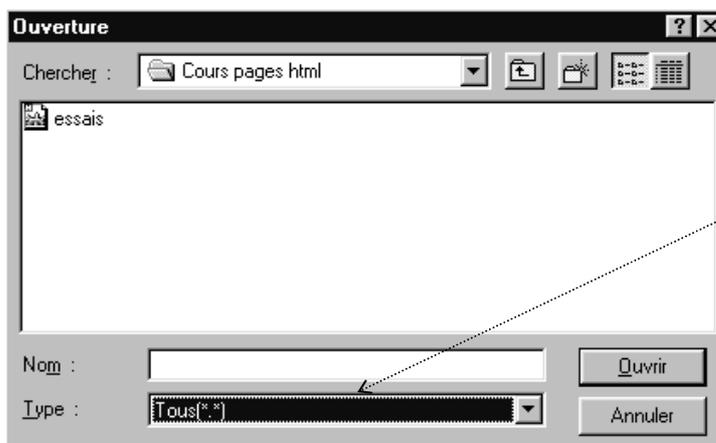
Fichier Nouveau - Ouvrir - Enregistrer



En général cet éditeur crée un format de fichier Word, mais pour nous Il faut créer un format de type texte



On enregistre un fichier HTML classiquement si ce n'est son suffixe .HTM et son type texte



On récupère un fichier HTML classiquement si ce n'est son suffixe .HTM accessible via la clé tous *.*

NAVIGATEURS

On parle encore de browsers ou de butineurs pour les outils permettant de se déplacer sur le Web

Deux entreprises monopolisent le marché, Firefox (selon les versions) et Microsoft avec son « explorer ».

Les deux produits offrent des caractéristiques semblables et chacun choisira...

Indépendamment du choix du produit, un certain nombre de manipulations peuvent se faire de façon semblable et nous permettrons de travailler

Une page Web étant disponible dans un fichier donné, on peut la charger sans aucune connexion, en local

Firefox

Pour charger un Fichier aller dans le Menu :

Fichier / Ouvrir un fichier... ou **CTRL + O**

Pour visualiser le source d'une page HTML, charger la page puis Menu :

Affichage / Code Source de la page ou **CTRL + U**

(les modifications sont impossibles)

Internet Explorer

Pour charger un Fichier aller dans le menu :

Fichier / Ouvrir ou **CTRL + O**

Pour visualiser le source d'une page HTML, charger la page: selon la version

Affichage / Source ouvre une fenêtre bloc note (modifiable)

Page / Afficher la Source ouvre une fenêtre bloc note (modifiable)



1° PAGE HTML ET TAGS

Construction

Dans l'éditeur, taper un document ayant l'aspect suivant :

```
<HTML>
<HEAD>
<TITLE>
première page Web
</TITLE>
</HEAD>

<BODY>
un premier essai de page Web en espérant que tout fonctionne !
</BODY>

</HTML>
```

L'enregistrer sur disque, puis l'afficher dans le navigateur en chargeant le fichier (si besoin, vous trouverez une copie sous le nom « **EXER00.HTM** »)

Pas de panique si l'aspect des caractères n'est pas le même que celui du document, en effet, aucune police n'étant définie, la police utilisée sera celle du navigateur par défaut, c'est à dire soit

Menu : **Option / Préférences générales / Police** sur Navigator

ou

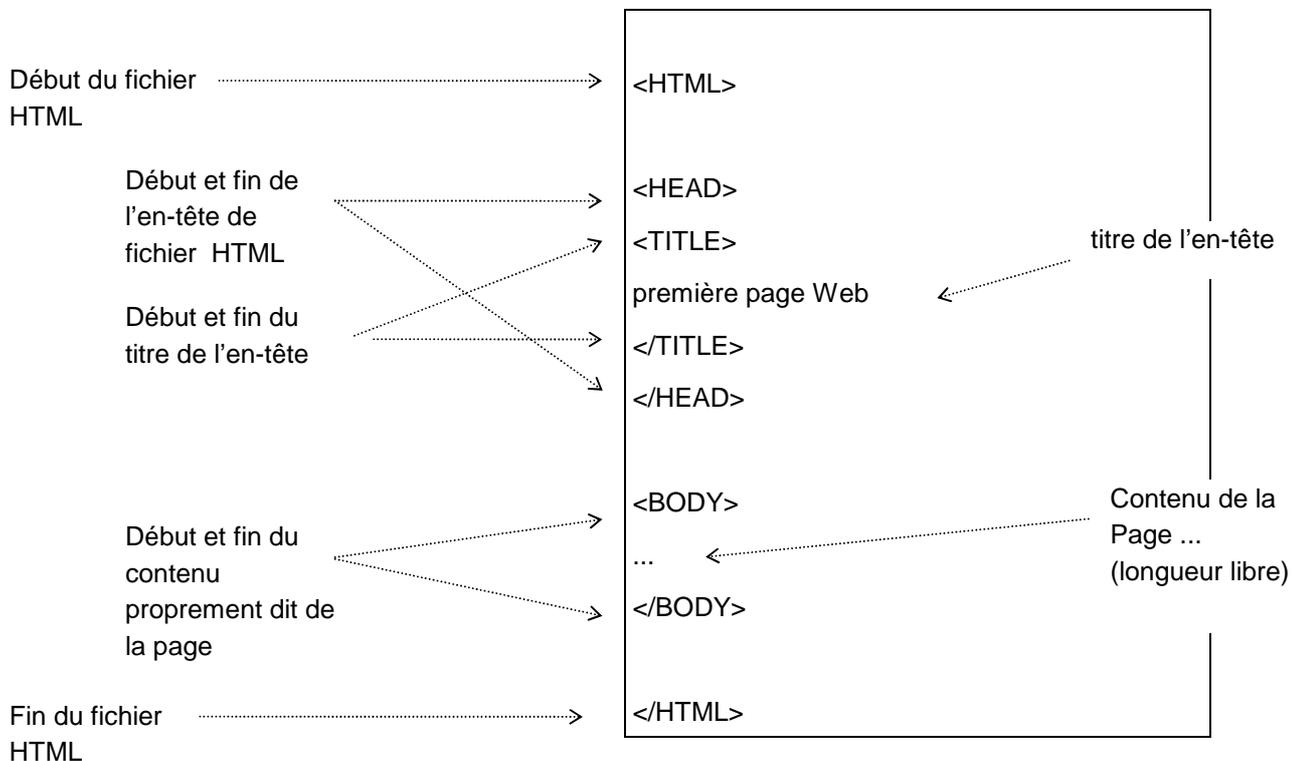
Menu : **Affichage / Option / Général / Affichage des polices** sur Explorer



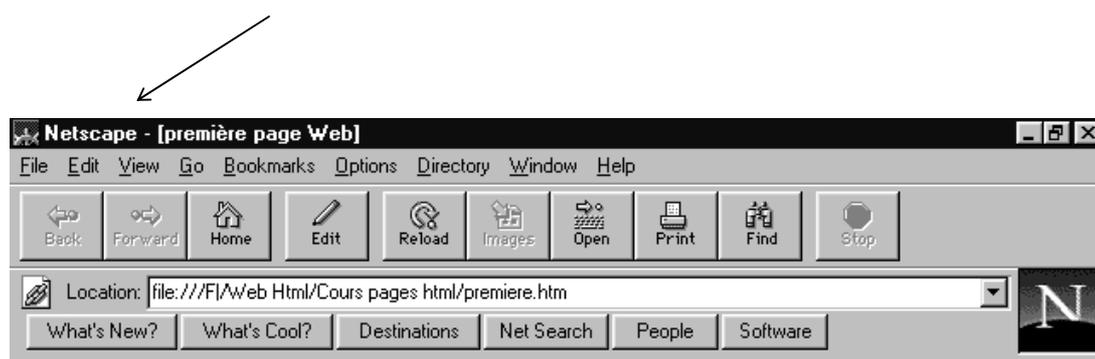
Explication des « tags » HTML

Une commande HTML est appelée tag

Voici le corps du document que nous avons tapé



Le titre sert à marquer une référence dans le navigateur



N.B: Dans les exemples suivants les fichiers HTML ne sont pas complets avec tous leurs tags. Pour un évident problème de place seule la partie faisant l'objet de notre attention est représentée

HTML ou XHTML

On peut voir souvent dans la littérature que à coté de HTML, d'autres sigles paraissent, et semblent de plus en plus utilisés de nos jours.

Le eXtensible HyperText Markup Language (XHTML) a été officialisé en 2000.

Il s'agit d'une reformulation du HTML sous les règles du eXtensible Markup Language (XML). Le XHTML demande un effort supplémentaire de rigueur, mais pour l'essentiel ressemble de très près au HTML. Un des très grands avantages de choisir d'utiliser le XHTML: puisque il est fondé sur le XML, tous les outils de traitement de XML deviennent d'un seul coup disponibles, permettant une variété impressionnante de manipulations permettant de reformater l'information concentrée dans le document.

Les différences fondamentales entre HTML et XHTML sont de l'ordre de la rigueur :

En XHTML les balises ouvrantes doivent être obligatoirement fermées

Exemple :

- en HTML on a une balise unique `<hr>` qui permet de créer une barre horizontale
- en XHTML on utilisera `<hr / >` pour obtenir la même chose

raisonnement identique pour la balise `
`

- en HTML on a une balise unique `
` qui permet un retour à la ligne sans fin de paragraphe
- en XHTML on utilisera `<br / >` pour obtenir la même chose

En XHTML les balises et propriétés sont écrites obligatoirement en minuscules

Exemple :

- en HTML on peut écrire la balise qui permet de créer une barre horizontale `<hr>` ou `<HR>` ou `<Hr>`
- en XHTML on utilisera obligatoirement la balise en minuscules `<hr / >` pour obtenir la même chose

En XHTML les valeurs sont écrites entre quotes (apostrophes) ou double quotes (guillemets)

Exemple :

- en HTML on peut écrire la balise qui permet de créer un titre centré `<h1 align=center >` ou `<h1 align="center" >`
- en XHTML on utilisera obligatoirement des quotes/doubles quotes `<h1 align="center" >` pour obtenir la même chose

En XHTML les valeur évoquées doivent correspondre à des propriétés

Exemple :

- en HTML on peut écrire `<input name="radiobutton" type="radio" value="radiobutton" checked>`
- en XHTML on écrira `<input name="radiobutton" type="radio" value="radiobutton" checked="checked" />`

En XHTML les balises doivent être correctement imbriquées

Exemple :

- En XHTML ``Ceci est un texte en gras et italique ``
On ferme les balises imbriquées dans le sens inverse de leur ouverture. La première balise ouverture est la dernière à se fermer. La dernière ouverte est la première à se fermer.

Le squelette d'une page XHTML - HTML

Puisqu'un document XHTML ne constitue qu'un cas particulier de document XML, on doit révéler au début du document quelles balises seront utilisées.

La déclaration `<!DOCTYPE>` sert justement à indiquer cela.

- Le **Document Type Definition** (DTD), qui précise au navigateur quelle version de XHTML vous utilisez :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

- ensuite, vous devez déclarer l'**espace de noms XML** (XMLNS) :

```
<html xmlns="http://www.w3.org/1999/xhtml">
```



Table des caractères

Un caractère peut-être aussi défini par son code numérique **é = é = é**

Donc une autre méthode existe et consiste à taper le code ASCII du caractère voulu précédé du #. Les caractères les plus fréquemment utilisés, avec la page ISO-8859-1 sont

eacute	pour	é	mais aussi	#193
Eacute		É		#201
egrave		è		#232
Egrave		È		#200
ecirc		ê		#234
ugrave		ù		#249
ucirc		û		#251
acirc		â		#226
agrave		à		#224
ocirc		ô		#244

Ainsi notre phrase lisible internationalement deviendrait

un premier essai de page Web en espérant que tout fonctionne !

Les caractères spéciaux

Il existe des caractères spéciaux non directement imprimables en HTML, respectivement

&lt;	pour	<
&gt;	pour	>
&amp;	pour	&
&quot;	pour	"

essayez d'incorporer dans votre page WEB quelques phrases du type

Société Brun père & fils

Centre d'Etude Universitaire "Pierre Mendes France"



Codage Unicode

Unicode est un système dans lequel les signes ou éléments de toutes les cultures d'écriture connues sont fixés. Par ce système, il devient possible de dire à un ordinateur quel signe on veut voir représenté. La condition préalable est naturellement que l'ordinateur ou le programme exécuté connaisse le système unicode. Chaque signe ou élément dans l'unicode est exprimé par un chiffre le standard Unicode encode les caractères sur plusieurs caractères (UTF-8, UTF-7, UTF-16, UTF-32). Dans la version 2.0 du standard unicode, 38885 signes sont documentés

À partir de la version 4.0 du langage, HTML utilise le jeu de caractères universel (Universal Character Set - UCS) d'après le standard ISO 10646. Ce standard se fonde de son côté sur la version 2.0 du système unicode.

Vous pouvez à partir du standard HTML 4.0 noter dans un fichier HTML des signes au choix extraits de l'UCS ou du système unicode avec la mention numérique correspondante. Comme dans:

ü <!-- lettre "ü" en décimal -->
ü <!-- lettre "ü" en hexadécimal -->

On pourra trouver sur <http://www.alanwood.net/> les tables UNICODES utilisables

- Standard Latin (jeu de caractères ASCII)

signes (en mention hexadécimale) de **U+0000 à U+007F**

→ **C0 Controls and Basic Latin** U+0000 – U+007F (0–127)

A	65	0041	LATIN CAPITAL LETTER A
B	66	0042	LATIN CAPITAL LETTER B
C	67	0043	LATIN CAPITAL LETTER C
D	68	0044	LATIN CAPITAL LETTER D
E	69	0045	LATIN CAPITAL LETTER E

- Latin-1 (voir aussi jeu de caractères iso 8559-1)

signes (en mention hexadécimale) de **U+0080 à U+00FF**

→ **C1 Controls and Latin-1 Supplement** U+0080 – U+00FF (128–255)

€ ¼ Ñ ñ

- Arabe

signes (en mention hexadécimale) de **U+0600 à U+06FF**

→ **Arabic** U+0600 – U+06FF (1536–1791)

؟ ب د ٣

N.B: Leur utilisation est plus délicate car ils sont moins bien reconnus par Internet Explorer 6. Notez que Mozilla depuis la version 1, Netscape depuis la version 6 les prennent tous en charge, et Amaya 8 tous sauf six

Unicode dépasse ainsi sensiblement le principe des différents jeux de caractères. À l'aide de ce système il est possible par exemple, en plein milieu d'un texte français de noter quelques mots en arabe

Ainsi si on veut la page

```
les caractères accentués sont utiles|
mais pourquoi pas aussi ceux là ? dans la même page ?
```

ث ت ة ب ا

Le code pourrait être (cf le fichier **iso-unicode.htm**)

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>Document sans nom</title>
</head>

<body>
<hr width="80%">
<p>les caractères accentués sont utiles</p>
<p>mais pourquoi pas aussi ceux là ? dans la même page ? </p>
<p><font size="6">ث ت ة ب ا</font></p>
```

pour information...

	1575	0627	ARABIC LETTER ALEF
ب	1576	0628	ARABIC LETTER BEH
ة	1577	0629	ARABIC LETTER TEH MARBUTA
ت	1578	062A	ARABIC LETTER TEH
ث	1579	062B	ARABIC LETTER THEH

Balise Méta charset =iso-xxxx

On peut insérer au début de la page HTML une balise Meta indiquant le jeu de code que l'on utilise

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
```

La mention du jeu de caractères est donc définie grâce à **http-equiv="content-type"**. On peut déjà y voir que la mention s'adresse aussi au serveur Web.

Pour **content=** mentionnez d'abord entre guillemets le type Mime qui est toujours **text/html** pour un fichier HTML.

Ensuite séparé par un point virgule suit la mention du jeu de caractères. Le jeu de caractères **iso-8859-1** qui a été défini est valable pour les langues d'Europe de l'ouest). Sont permises les mentions de jeux de caractères mentionnées à l'adresse Web

<http://www.iana.org/assignments/character-sets>.



Les espaces et les fins de ligne

On a déjà vu le problème de la police par défaut. Mais il faut savoir que en fonction de la résolution des écrans et de la taille des fenêtres, les fichiers HTML peuvent se présenter de manière très diverse

Tout navigateur coupe automatiquement les lignes pour afficher l'ensemble du texte, et uniquement à l'occasion d'espaces. De plus il convertit tout retour à la ligne dans le source en espace, et remplace plusieurs espaces successifs par un seul espace.

Par conséquent les sauts de lignes n'ont pas à être "prévus", de même qu'ils peuvent être insérés dans la page Web pour en faciliter la lecture lors de sa construction, cela n'aura aucune influence sur l'affichage dans le navigateur !

tapez ces 3 paragraphes ci-dessus de façon à être affichables en HTML

```
<HTML>
<HEAD>
<TITLE>                                Fichier EXER01.HTM
exercice 1
</TITLE>
</HEAD>
<BODY>

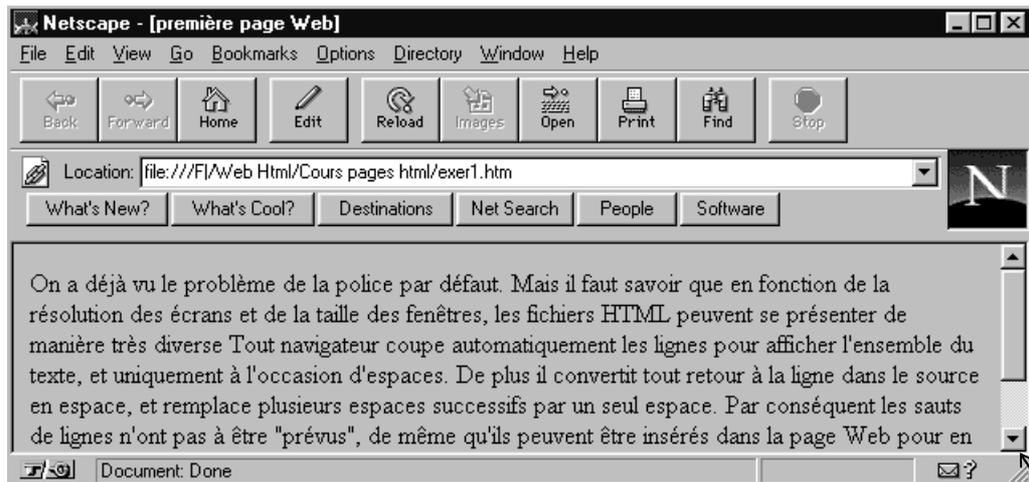
On a déjà vu le problème de la police par défaut.

Mais il faut savoir que en fonction de la résolution des écrans et de
la taille des fenêtres, les fichiers HTML peuvent se présenter de
manière très diverse

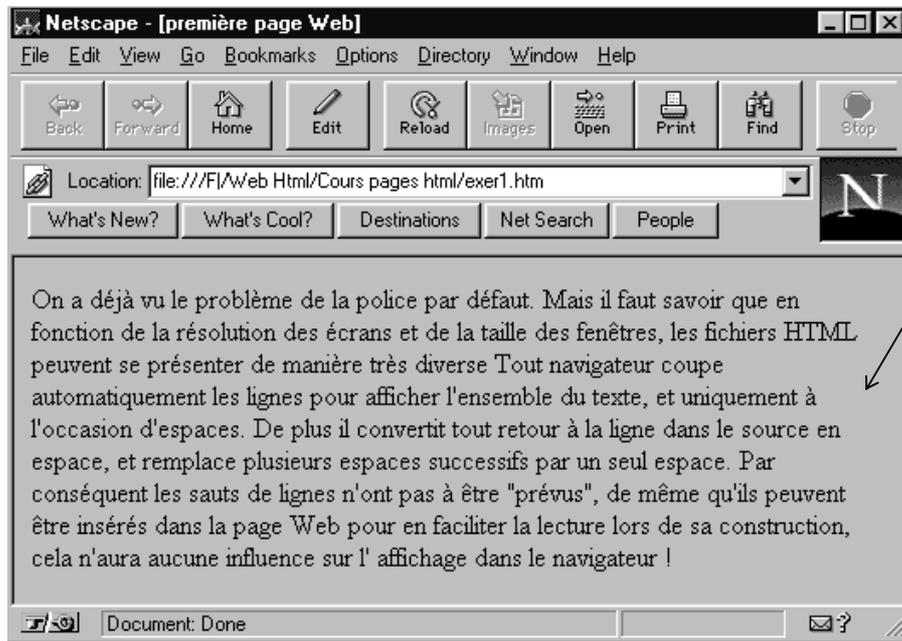
Tout navigateur coupe automatiquement les lignes pour afficher l'ensemble du
texte, et uniquement à l'occasion d'espaces. De plus il convertit tout retour
à la ligne dans le source en espace, et remplace plusieurs espaces
successifs par un seul espace.

Par conséquent les sauts de lignes n'ont pas à être
prévus", de même qu'ils peuvent être insérés dans la page Web pour en faciliter la lecture lors
de sa construction, cela n'aura aucune influence sur l'affichage dans le
navigateur !

</BODY>
</HTML>
```



En fonction de la dimension de la fenêtre...



Les retours à la ligne et les barres de défilement sont gérées

On ne peut par conséquent jamais savoir sur combien de lignes notre page va être affichée

MISE EN FORME DE TEXTE 1°

Saut de ligne - taille - gras - italique

A partir d'un texte HTML correct, c'est à dire construit avec les caractères spéciaux nécessaires, on veut pouvoir le présenter

Les tags nécessaires sont les suivants

<code>
</code>		provoque un saut de ligne
<code><P></code>	<code></P></code>	provoque un saut de ligne suivi de l'insertion d'une ligne vierge (fin de paragraphe) NB: la répétition de ce tag n'entraîne pas plusieurs ligne blanche, mais une seule ! NB: le tag de fermeture <code></P></code> est optionnel mais conseillé
<code><Hx></code>	<code></Hx></code>	Crée un titre et modifie la taille du texte compris entre les deux tags selon une échelle variant de 1 (+grand) à 6 (plus petit) Insère de plus une ligne blanche après (équivalent à une balise <code><P></code> avec mise en évidence en plus)
<code></code>	<code></code>	met en gras (Bold) peut s'insérer même autour d'un caractère dans un mot
<code><I></code>	<code></I></code>	met en italique peut s'insérer même autour d'un caractère dans un mot

Vous pouvez regarder le fichier **EXER02.HTM** et noter les effets des tags employés

```
<BODY>
```

Tout navigateur coupe automatiquement les lignes pour afficher l'ensemble du texte, et uniquement ` l'occasion d'espaces. De plus il convertit tout retour ` la ligne dans le source en espace, et remplace plusieurs espaces successifs par un seul espace.

```
<H1> Compris ? </H1>
```

```
<H2> Compris ? </H2>
```

```
<H3> Compris ? </H3>
```

```
<H4> Compris ? </H4>
```

```
<H5> Compris ? </H5>
```

```
<H6> Compris ? </H6>
```

```
Compris ? <P>
```

```
</BODY>
```

Noter que lors de l'utilisation des Hx un saut de ligne est toujours créé, sans avoir besoin du tag <P>

et cela donnerait

Compris ?

présentez le texte suivant (solution en **EXER03.HTM**)

Bienvenue sur ma *page d'accueil* !

Je m'appelle **Michel Cabaré**.

Parallèlement à ma passion de *l'informatique*, j'aime les voyages et la photographie. Parmi mes hobbies, je citerai également la musique et le "footing".

Revenez me rendre visite sur ma *page d'accueil*

Cabare@cuefa.fr

RETRAITS ET LISTES

Retraits

Les pages HTML servent à présenter des informations, et très souvent on a besoin de mettre ces informations en retrait ou de faire des énumérations, qu'elles soient numérotées ou non

Les tags nécessaires sont les suivants

<code><DL></code>	<code></DL></code>	Data List: marqueur de début et de fin de liste encadre généralement des tag DT et DD peut inclure des sous listes marquées elles-mêmes par <code><DL></code> et <code></DL></code>
<code><DT></code>	<code></DT></code>	Data Term: marqueur de début et de fin d'entrée dans une liste sans retrait un saut de ligne est effectué après
<code><DD></code>	<code></DD></code>	Data Definition: marqueur de début et de fin d'entrée dans une liste avec retrait un saut de ligne est effectué après

Ainsi si on veut obtenir l'écran suivant

Bienvenue au C.U.E.F.A
Centre Universitaire Enseignement Formation Adulte

Notre comité

a. Président :
 Mélanie Zétofrais

b. Responsable Formation Continue :
 Quentin Chaquin

Secrétariat :
 Catherine Neithe

il faut taper le fichier HTML suivant (**EXER04.HTM**)

Ce qui donnerait à peu près ceci

Début et fin de Liste

```
...
<TITLE>
  C.U.E.F.A.
</TITLE>
<HEAD>

<BODY>
<H3>
  Bienvenue au C.U.E.F.A
  <BR>
  Centre Universitaire Enseignement Formation Adulte
</H3>
<B>Notre comit&eacute;</B>
<DL>
  <DT>a. Pr&eacute;sident :</DT>
  <DD>M&eacute;lanie Z&eacute;tofra<sup>s</sup></DD>
  <DT>b. Responsable Formation Continue :</DT>
  <DD>Quentin Chaquin</DD>
  <DT>Secr&eacute;tariat :</DT>
  <DD>Catherine Neithe</DD>
</DL>
...
```

DT sans retrait...

DD avec retrait...

Que faudrait-il faire pour obtenir ceci ? (décalage de secrétariat)

```
Bienvenue au C.U.E.F.A
Centre Universitaire Enseignement Formation Adulte

Notre comité

a. Président :
  Mélanie Zétofra<sup>s</sup>
b. Responsable Formation Continue :
  Quentin Chaquin
Secrétariat :
  Catherine Neithe
```

la solution est en **EXER04B.HTM** et repose sur l'imbrication de deux listes

Listes non numérotées

L'idée est de présenter une série de points ou de notions chacune mise en évidence par le biais d'une puce. L'imbrication de sous listes est tout à fait autorisée

Les tags nécessaires sont les suivants

<code></code>	<code></code>	Unnumbered List: marqueur de début et de fin de liste non numérotée; un saut de ligne avec ligne blanche est inséré avant et après encadre généralement des tag LI peut inclure des sous listes marquées elles-mêmes par <code></code> et <code></code>
-------------------------	--------------------------	--

Ce tag peut être modifié par la mention `TYPE="DISC","CIRCLE","SQUARE"` qui modifie la forme de la puce

<code></code>	<code></code>	List Item : annonceur d'une entrée de liste qui sera dotée d'une puce avec un retrait et un saut de ligne
-------------------------	--------------------------	--

Listes numérotées

L'idée est de présenter une série de points ou de notions chacune mise en évidence par un numéro d'ordre donné automatiquement

Les tags nécessaires sont les suivants

<code></code>	<code></code>	Ordered List: marqueur de début et de fin de liste numérotée; un saut de ligne avec ligne blanche est inséré avant et après encadre généralement des tag LI peut inclure des sous listes marquées elles-mêmes par <code>...</code> ou <code>...</code>
-------------------------	--------------------------	--

Ce tag peut être modifié par la mention `TYPE="a", "l", "i" "1"` qui modifie le style de numérotation (Netscape)

<code></code>		List Item : annonceur d'une entrée de liste qui sera dotée d'un numéro avec un retrait et un saut de ligne
-------------------------	--	---

En résumé on dispose de trois types de liste :

Liste avec Retrait	Liste à Puces	Liste Numérotées
<DL> (début de liste)	 (début de liste)	 (début de liste)
<DT> </DT> (sans retrait)	 (entrée)	 (entrée)
<DD> </DD> (avec retrait)	 (entrée)	 (entrée)
</DL> (fin de liste)	 (fin de liste)	 (fin de liste)

```

Formations dispensées
1. Anglais
2. Informatique
   o bureautique
     1. Word
     2. Excel
   o gestion
     1. Access
     2. Html
     3. Saari Comptabilité
   o Industrielle
3. Mathématiques
    
```

A partir de ce fichier
(en **EXER05.HTM**)

Construisez cet exemple
(en **EXER05b.HTM**)

```

Bienvenue au C.U.E.F.A
Centre Universitaire Enseignement Formation Adulte

Notre comité

Président :
    Mélanie Zétofrais
Responsable Formation Continue :
    Quentin Chaquin
Secrétariat :
    Catherine Neithe

1. Anglais
2. Informatique
   ■ Bureautique
     1. Word
     2. Excel
   ■ Gestion
     1. Access
     2. Html
     3. Saari Comptabilité
   ■ Industrielle
3. Mathématiques
    
```

INSÉRER DES LIGNES

La ligne sert à améliorer la présentation des pages HTML

Les tags nécessaires sont les suivants

<HR> écrit une ligne horizontale (Horizontal Rule) et provoque un saut de ligne avec un espacement avant et après

Ce tag peut être modifié par le paramètre SIZE=X (X compris entre 1 et ...) où X indique en pixel la hauteur de la ligne si absent la taille sera 2 pixels

Ce tag peut être modifié par le paramètre ALIGN=X (X valant "right", "left" ou "center") si absent la ligne sera centrée

Ce tag peut être modifié par le paramètre WIDTH=X (X valant soit un entier représentant des pixels soit un % de largeur d'écran exprimé entre guillemets) si absent la ligne vaudra "100%" de la largeur d'écran

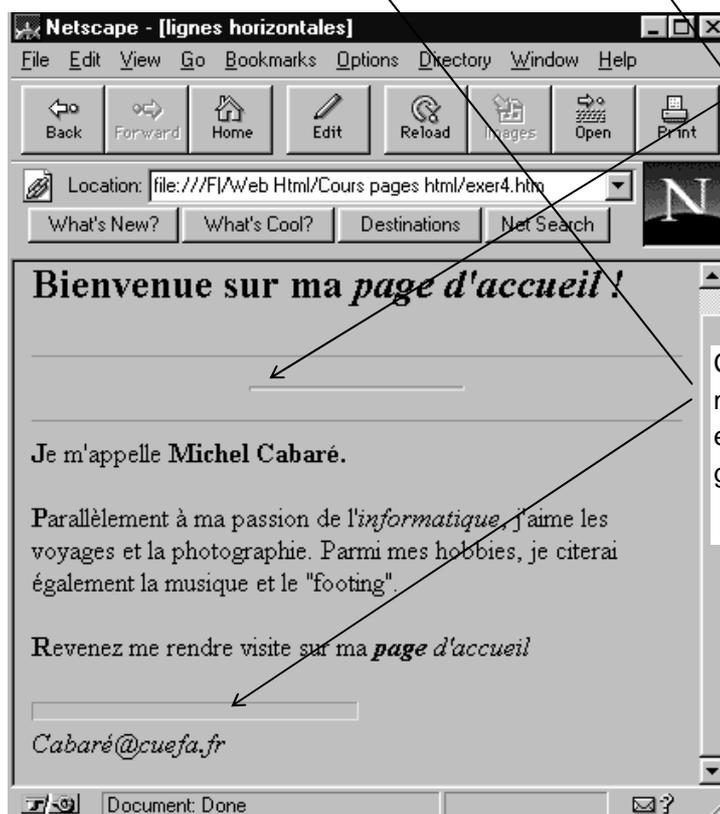
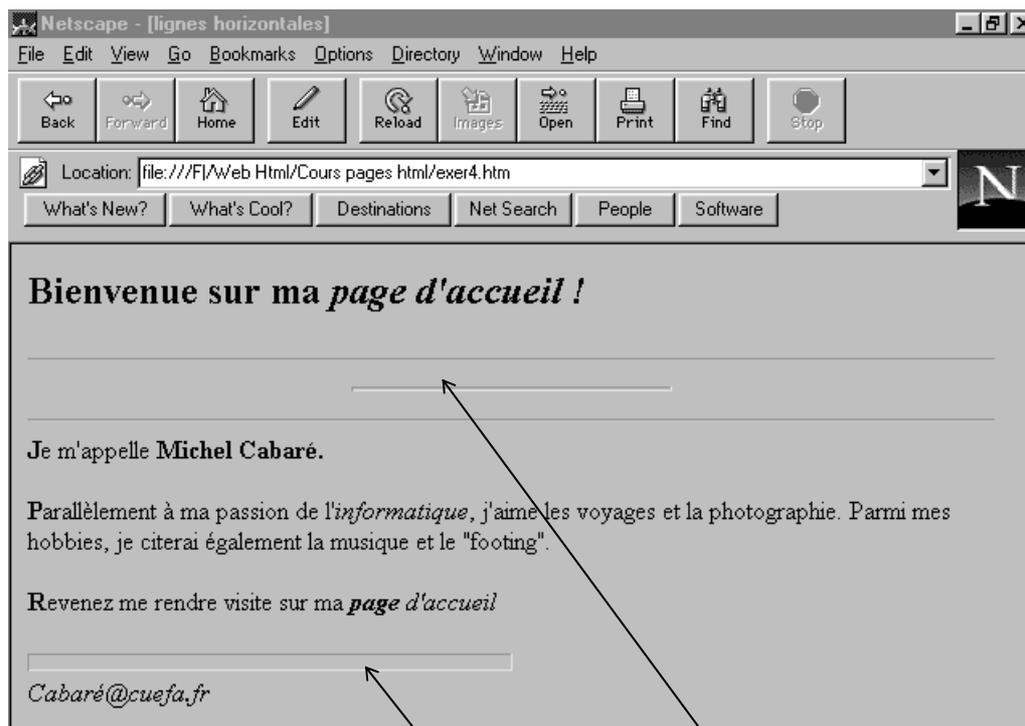
Il est préférable d'utiliser la méthode % qui s'affranchit de la résolution de l'écran et de la taille de la fenêtre

Ce tag peut être modifié par la mention NOSHADE qui empêche de tracer l'ombre portée du trait

Ainsi le Tag <HR>
équivalent à <HR SIZE=2 ALIGN="center" WIDTH="100%">

N.B : une variation de ce tag autorise la coloration de la ligne mais n'est pas reconnue par tous les navigateurs...

Essayez de réaliser la page suivante de façon à ce qu'elle garde ses traits même si la fenêtre du navigateur se voit modifiée (**EXER06.HTM**)



Cette ligne fait 1/3 de l'écran et elle est centrée

Cette ligne fait la moitié de l'écran et elle est alignée à gauche

MISE EN FORME DE TEXTE 2°

Taille de texte

on a déjà vu comment faire varier la taille de titres, avec 6 niveaux (H1,H2...), mais on peut modifier la taille d'un texte sans pour autant que le tag ne génère un saut de ligne avec une ligne blanche

Les tags nécessaires sont les suivants

`` `` définit la portion de texte qui doit avoir une taille précisée par la valeur x, avec x variant de 1 (plus petit) à 7 (plus grand) par défaut la taille vaut 3

N.B: On peut trouver parfois une notation de x en relatif +2 ou -1 par exemple, la taille se référant alors par rapport à celle du texte précédant, à déconseiller!

Couleur

Tous les textes s'affichent en noir mais on peut remédier à la chose

Les tags nécessaires sont les suivants

`` `` défini la portion de texte qui doit avoir une couleur définie par x valant "#000000" en hexadécimal. Dans cette valeur on a le Rouge, le Vert et le Bleu pouvant chacun avoir une palette de 256 nuances

Ainsi ce codage en hexadécimal permet de "lire" la teneur respectivement en Rouge Vert et Bleu du texte avec 00 mini et FF maxi (255).

Par exemple : 000000 est noir FF0000 est rouge
 00FF00 est vert 0000FF est bleu
 FFFFFFFF est blanc

Chargez le fichier **Couleur.htm** dans le navigateur



Le code d'une couleur en HTML s'écrit donc ainsi :
"#33CC66"

Trois paires de "chiffres" 00, 33, 66, 99, CC, FF, après un dièse #.

Mais une série de mots clés permet de définir également les couleurs sans passer par leur valeur hexadécimale

Voici un tableau récapitulant (par nom) les 16 couleurs reconnues selon la norme **HTML 3.2**.

Nom	Couleur affichée
BLACK	Noir
MAROON	Marron ou rouge foncé
GREEN	Vert
OLIVE	Vert olive
NAVY	Bleu marine
PURPLE	Violet
TEAL	Cyan foncé
GRAY	Gris
SILVER	Argent
RED	Rouge
LIME	Vert clair
YELLOW	Jaune
BLUE	Bleu
FUCHSIA	Fuschia
AQUA	Bleu clair
WHITE	Blanc

Quelques couleurs reconnues uniquement avec les versions 4 ou 5 des navigateurs.

Nom	Couleur affichée
AZURE	Bleu azur
BISQUE	Beige
BROWN	Brun
BLUEVIOLET	Bleu-violet
CHARTREUSE	Vert clair
CHOCOLATE	Brun clair
CORNSILK	Rose clair

DARKGREEN	Vert foncé
DARKORANGE	Orange foncé
DARKORCHID	Mauve foncé
DEEPSKYBLUE	Bleu ciel
GOLD	Doré
IVORY	Ivoire
ORANGE	Orange
LAVENDER	Lavande
PINK	Rose
PLUM	Prune
SALMON	Saumon
SNOW	Neige
TURQUOISE	Turquoise
WHEAT	Jaune paille

Et encore

<code><STRIKE></code>	<code></STRIKE></code>	définit la portion de texte qui sera barré
<code><BLINK></code>	<code></BLINK></code>	définit la portion de texte clignotante
<code><SUB></code>	<code></SUB></code>	définit la portion de texte en indice
<code><SUP></code>	<code></SUP></code>	définit la portion de texte en exposant
<code><CENTER></code>	<code></CENTER></code>	définit la portion de texte qui sera centrée

On peut noter que la présence de plusieurs paramétrages est possible dans un seule instruction de type FONT comme dans

```
<FONT SIZE=4 COLOR="#FF0000">
```

dans ce cas le tag `` annule les deux effets

mais si plusieurs instructions FONT existent et se suivent, le tag d'annulation ne s'applique qu'au tag de commande immédiatement précédant

présenter la page d'accueil du Cueva avec ces nouveaux tags

LES TABLEAUX

Construction

Les tableaux ont été "introduits de force" par Netscape dans la norme HTML en Avril 95 et sont désormais reconnus depuis la version 2.0

Ils sont très importants car c'est sur eux que repose une bonne partie des astuces de présentation des pages HTML sophistiquées

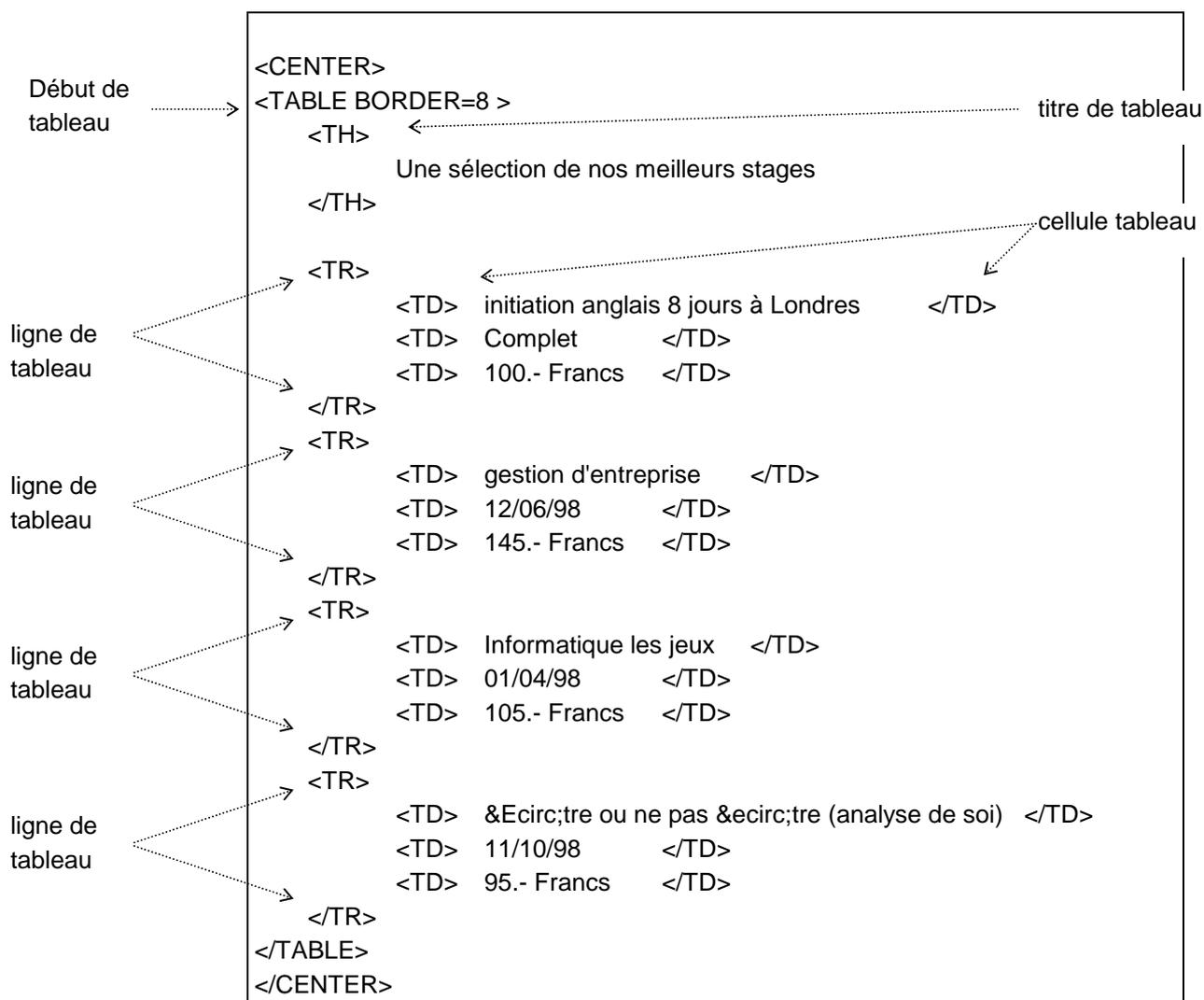
Les tags nécessaires sont les suivants

<TABLE >	</TABLE>	marqueur de début et de fin de tableau
		Ce tag peut être modifié par le paramètre BORDER=X (X varie entre 0 et ...) où X indique en pixel la hauteur des lignes utilisées pour tracer le tableau.
<TR>	</TR>	marqueur de début et de fin de ligne de tableau
<TD>	</TD>	marqueur de début et de fin de texte d'une cellule
		Le nombre de ces couples définira le nombre de colonnes du tableau
<TH>	</TH>	marqueur du titre du tableau qui est automatiquement centré et affiché en gras

Si on veut par conséquent obtenir un tableau de ce type (**EXER07.HTM**)

Une sélection de nos meilleurs stages		
initiation anglais 8 jours a londres	Complet	100.- Francs
gestion d'entreprise	12/06/98	145.- Francs
Informatique les jeux	01/04/98	105.- Francs
Être ou ne pas être (analyse comportementale)	11/10/98	95.- Francs

le fichier HTML aurait la structure suivante :



Notez que la présentation du fichier HTML compte énormément pour la lisibilité et la compréhension

Présentation

Le tableau ainsi présenté, souffre de plusieurs défauts corrigibles via quelques paramétrages des tags que l'on vient d'étudier

Les tags

<TABLE > et <TD> acceptent un paramètre **WIDTH**

Ces tags peuvent être modifiés par le paramètre WIDTH=X avec X valant soit un entier représentant des pixels soit un % de largeur d'écran (exprimé entre guillemets) si absent la ligne vaudra "100%" de la largeur d'écran

N.B: si "100%" occupe tout l'écran une valeur supérieure génère une barre de défilement

<TABLE > accepte les paramètres **CELLSPACING** et **CELLPADDING**

Ce tags peut être modifié par le paramètre CELLSPACING=X avec X valant un entier représentant en pixels l'espacement entre les bordures de cellule

Ce tags peut être modifié par le paramètre CELLPADDING=X avec X valant un entier représentant en pixels l'espacement entre le texte d'une cellule et la bordures de cette même cellule

<TD > accepte le paramètre **ALIGN**

Ce tags peut être modifié par le paramètre ALIGN=X avec X valant "center" ou "right" et modifiant l'alignement du contenu de la cellule

Si omis, l'alignement gauche est adopté

<TH > et <TD> acceptent le paramètre **COLSPAN**

Ces tags peuvent être modifiés par le paramètre COLSPAN=X avec X valant le nombre de cellules (colonnes) sur lesquelles le titre s'étendra

<TD > accepte les paramètre **ROWSPAN**

Ce tags peut être modifié par le paramètre ROWSPAN=X avec X indiquant le nombre de lignes à fusionner

A l'aide de ces nouveaux paramètres présenter le tableau sous l'aspect suivant (**EXER08.HTM**)

COLSPAN=3

CELLSPACING=8

CELLPADDING=12

BORDER=20

Une sélection de nos meilleurs stages		
initiation anglais 8 jours a londres	Complet	
gestion d'entreprise	12/06/98	145.- Francs
Informatique les jeux	01/04/98	105.- Francs
Être ou ne pas être (analyse comportementale)	11/10/98	95.- Francs

et encore

des tableaux peuvent être imbriqués, il suffit de préparer le code d'un tableau puis de l'insérer dans une cellule

De même des listes peuvent être incluses dans un tableau suivant le même principe

Quelques tags peuvent encore être utilisés

`<CAPTION>` `</CAPTION>` marqueur de début et de fin de présentation de tableau, c'est à dire d'un titre hors tableau mais parfaitement centré sur lui. Doit être placé entre `<Table>` et la première rangée

Ce tag peut être modifié par le paramètre `ALIGN=TOP` ou `BOTTOM` selon que l'on veut le placer juste avant ou après le tableau

Ainsi que quelques paramètres

`<TH >` et `<TD>` acceptent le paramètre `NOWRAP`

Ces tags peuvent être modifiés par le paramètre `NOWRAP` qui empêche que le contenu d'une cellule ne soit brisé si le navigateur le souhaite (des barres de défilement apparaîtront)

`<TH >` et `<TD>` acceptent le paramètre `VALIGN`

Ces tags peuvent être modifiés par le paramètre `VALIGN` qui permet d'aligner verticalement le contenu par la valeur "top", "middle" "bottom" ou bien sur une même ligne pour tous par "baseline"

Mise en forme

Dans un tableau, toute mise en forme, et notamment la couleur éventuelle du texte doit être spécifiée pour chaque cellule

Par exemple la couleur est modifiable via le paramètre

`BGCOLOR = xxxxx`

mais la signification dépendra de la balise dans laquelle on l'incorpore

`<TABLE BGCOLOR="FFFFFF"...>` donne la couleur du tableau

`<TD BGCOLOR="FFFFFF"...>` donne la couleur de la cellule

Faites le tableau suivant (solution en **EXER08b.HTM**)

Une sélection de nos meilleurs stages		
initiation anglais 8 jours a londres	Complet	
gestion d'entreprise	12/06/98 semaine	145.- Francs
Informatique les jeux	01/04/98	gratuit
Être ou ne pas être (analyse comportementale)	11/10/98	

ainsi que le tableau suivant (solution en **EXER08c.HTM**)

Une sélection de nos meilleurs stages		
initiation anglais 8 jours a londres	Complet	
gestion d'entreprise	12/06/98 semaine	145.- Francs
Informatique les jeux	01/04/98	gratuit
Être ou ne pas être (analyse comportementale)	11/10/98	250.- Francs
	12/10/98	

POUR UN FICHER HTML RELISIBLE

Commentaires

A partir du moment ou on construit des pages HTML, ces pages vont être modifiées. Plus leur nombre augmente, plus leur présentation est soignée et plus le source devient difficilement lisible à terme. On peut appliquer plusieurs méthodes pour que les fichiers HTML soient plus faciles à manipuler

on peut englober des commentaires dans la page qui seront ignorés par les navigateurs mais qui pourront servir comme repère

Les tags nécessaires sont les suivants

<!-- --> marqueur de début et de fin de commentaire

Disposition

Ne pas hésiter sur les sauts de ligne, ainsi que sur l'indentation

```
<HTML>
<!-- document écrit par michel pour le etc etc ->
  <HEAD>
    <TITLE>
      première page Web
    </TITLE>
  </HEAD>
  <BODY>
    ...
  </BODY>
</HTML>
```

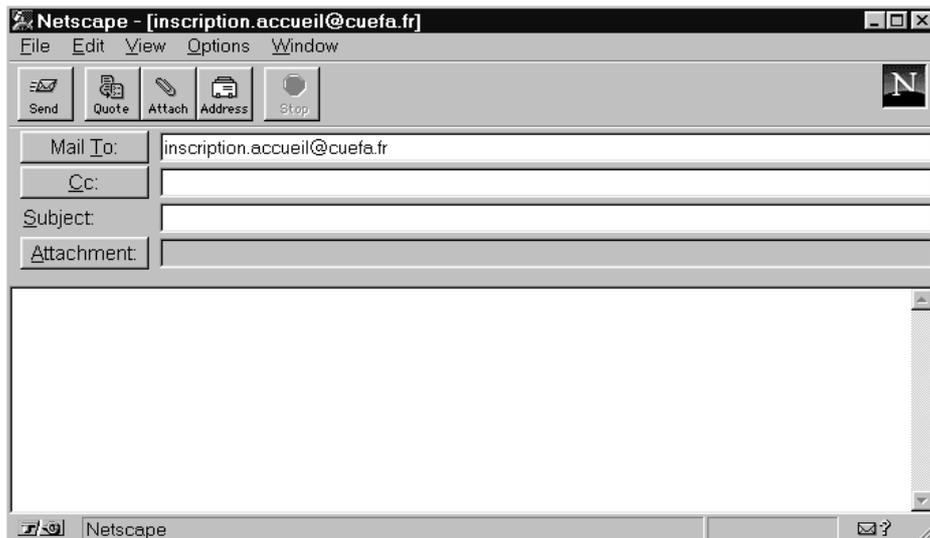
Lors de liste éventuelles
représenter si possible le
décalage...
idem pour les tableaux ...

Adresse mail

Pour référencer une adresse postale de type Email il suffit d'insérer une ligne du type

HREF="mailto:xxx"

pour que l'utilisateur puisse envoyer un mail simplement en cliquant sur un mot

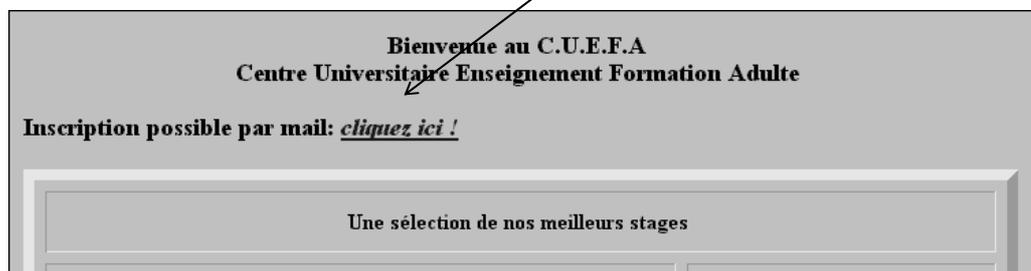


rajouter une inscription par mail à l'adresse fictive suivante

inscription.accueil@cuefa.fr

présentée ainsi (**EXER09.htm**)

Aspect Italique et
clignotant de "cliquez ici"



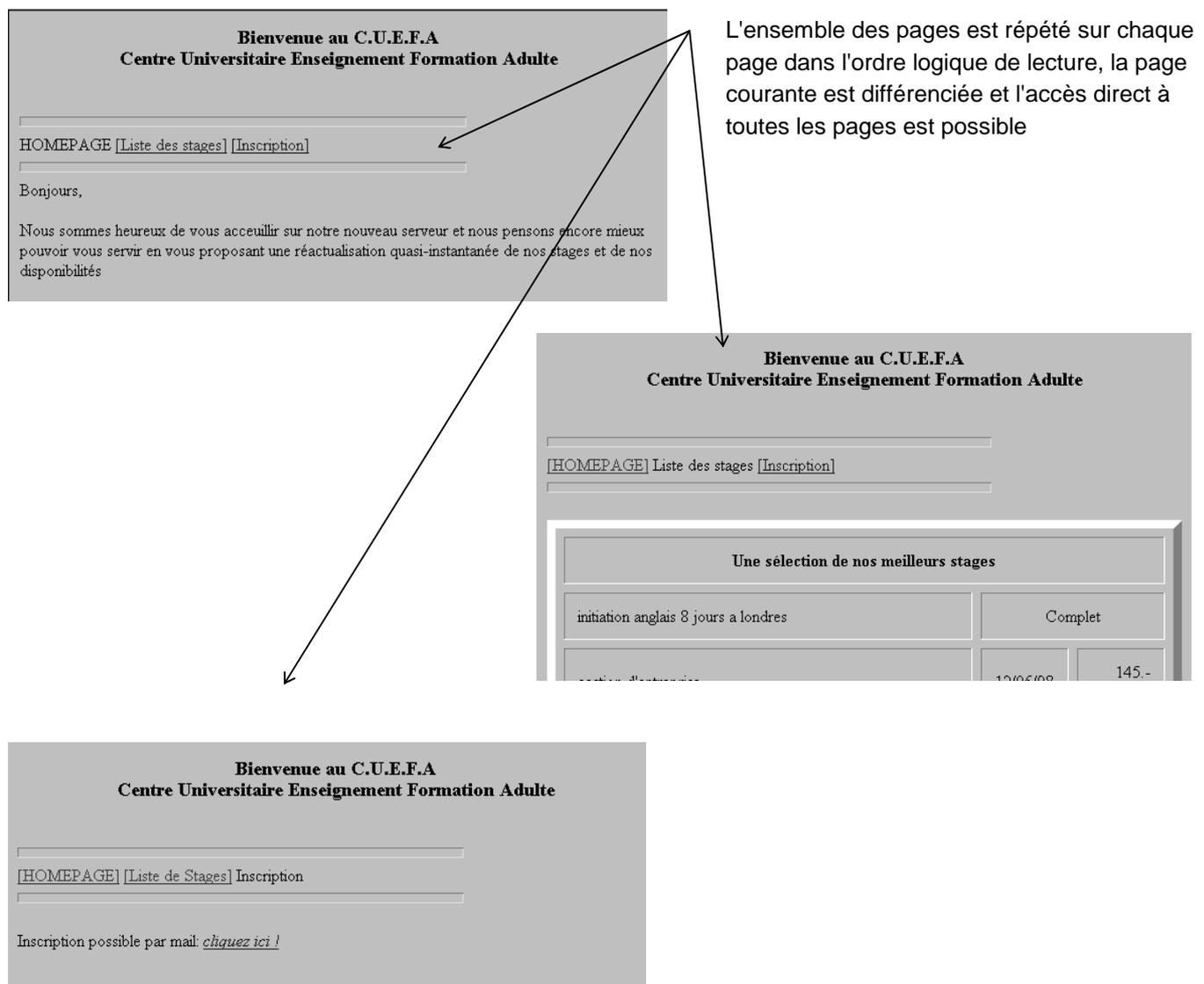
Liens entres pages HTML

Toute création de site sur le WEB repose nécessairement sur la production de "pages" composant les informations que vous allez diffuser

L'architecture de ces pages avec la navigation que l'utilisateur pourra effectuer entre elles n'est pas toujours des plus simples, surtout parce que le lecteur ne voit qu'une page à la fois (et parfois moins)

Par conséquent il faut lui donner toujours le moyen de se repérer dans l'ensemble du site sans qu'il ait l'impression gênante de tourner en rond

1. Dans une optique de repérage le lecteur doit toujours savoir où il se trouve par rapport à l'ensemble des pages du site :



Comment obtenir cette construction ?

Le problème ne se pose pratiquement pas tant que toutes les pages requises se trouvent dans le même répertoire, car alors le lien s'écrirait simplement

```
<A HREF="exer10_1.htm">[Stages]</A>
```

Ainsi le découpage de notre page précédente en 3 pages nouvelles, une pour l'accueil, une pour la liste des stages et une pour les inscriptions amènerait les modifications suivantes dans la page d'accueil (**exer10_1.htm**)

```

<TITLE>
  C.U.E.F.A. Homepage
</TITLE>
<H3>
<CENTER>
Bienvenue au C.U.E.F.A
<BR>
Centre Universitaire Enseignement Formation Adulte
</CENTER>
</H3>

<BR>
<HR SIZE=10 WIDTH = "70%" ALIGN="left">
HOMEPAGE
<A HREF="exer10_2.htm">[Liste des stages]</A>
<A HREF="exer10_3.htm"> [Inscription]</A>
<HR SIZE=10 WIDTH = "70%" ALIGN="left">

Bonjour,<P>

```

Rappel "pour ordre" de la page en cours

Références des pages que l'on peut atteindre

Titre de page apparaissant dans le navigateur rappelant la page visitée

Traçage d'une ligne horizontale au dessus et dessous des références

et les deux autres pages, indépendamment de leur contenu, porterait les éléments suivants:

```

<HR SIZE=10 WIDTH = "70%" ALIGN="left">
<A HREF="exer10_1.htm">[HOMEPAGE]</A>
Liste des stages
<A HREF="exer10_3.htm"> [Inscription]</A>
<HR SIZE=10 WIDTH = "70%" ALIGN="left">

```

```

<HR SIZE=10 WIDTH = "70%" ALIGN="left">
<A HREF="exer10_1.htm">[HOMEPAGE]</A>
<A HREF="exer10_2.htm">[Liste des stages]</A>
Inscription
<HR SIZE=10 WIDTH = "70%" ALIGN="left">

```

vous pouvez partir des 3 fichiers de départ bruts sans liens :

EXER10_1d,.HTM, EXER10_2d.HTM, EXER10_3d.HTM

(solution en : **EXER10_1.HTM, EXER10_2.HTM, EXER10_3.HTM**)

2. Dans un soucis de commodité le lecteur doit toujours pouvoir continuer la lecture "naturellement" :

"L'ensemble des pages est lié dans un ordre logique par des boutons suivant - précédent"

Bienvenue au C.U.E.F.A
Centre Universitaire Enseignement Formation Adulte

[HOMEPAGE] [Liste des stages] [Inscription]

Bonjours,

Nous sommes heureux de vous accueillir sur notre nouveau serveur et nous pensons encore mieux pouvoir vous servir en vous proposant une réactualisation quasi-instantanée de nos stages et de nos disponibilités

[suivant]

1°Page...

Bienvenue au C.U.E.F.A
Centre Universitaire Enseignement Formation Adulte

[HOMEPAGE] [Liste des stages] [Inscription]

Une sélection de nos meilleurs stages

initiation anglais 8 jours a londres	Complet	
gestion d'entreprise	12/06/98	145.- Francs
Informatique les jeux	01/04/98	105.- Francs
Être ou ne pas être (analyse comportementale)	11/10/98	95.- Francs

[précédant] [suivant]

2°Page...

Bienvenue au C.U.E.F.A
Centre Universitaire Enseignement Formation Adulte

[HOMEPAGE] [Liste de Stages] [Inscription]

Inscription possible par mail:

[précédant]

3°Page...

Visualisation en : **EXER10_1b.HTM, EXER10_2b.HTM, EXER10_3b.HTM**

Emplacement des pages HTML : relatif ou absolu ?

On l'a compris, il peut y avoir toute une arborescence de documents HTML reliés entre eux, stockés dans des répertoires différents du disque de la machine hôte

On peut référencer un lien de manière relative ou absolue

- de manière relative, tout appel est indexé sur le répertoire de la page active

<code>[nompag]</code>	même répertoire
<code>[nompag]</code>	répertoire parent
<code>[nompag]</code>	deux niveaux dessus
<code>[nompag]</code>	trois niveaux dessus
<code>[nompag]</code>	sous répertoire "toto"
<code>[nompag]</code>	sous sous répertoire "titi" dans sous répertoire "toto" du répertoire actuel

ainsi **../** ou **../../** ou **(etc)** permet de remonter l'arborescence

ainsi **toto/** ou **titi/toto/** ou **(etc)** permet de descendre dans l'arborescence en nommant les répertoire de destination

- de manière absolue, tout appel est indexé sur le disque choisi

`[nompag]`

ainsi **file:///** suivit de **unité:/dossier/sous-dossier/** **etc**
permet de repérer un fichier sur une machine

Attention :

en HTML il faut utiliser le **/** (slash ou barre oblique) non pas le **** (anti-slash ou contre-slash)

Paramétrage de l'emplacement

En rajoutant dans l'en-tête d'un document le tag

```
<BASE HREF="//devellope/exercice/">
```

on sait que toutes les références placées dans ce document comme

```
<HREF="images/ampoule.gif">
```

seront en fait décodées comme

```
<HREF="//devellope/exercice/images/ampoule.gif">
```

Structure linéaire ou hiérarchique

- Dans une structure linéaire, on a une présentation directement inspirée de celle d'un livret de quelques pages

Il s'agit de structurer le site comme on l'a vu, chaque page portant en haut un bandeau rappelant l'ensemble des pages et permettant d'y accéder directement, plus une table des matières et un index contenant des points d'entrées précis dans ces pages.

HOME PAGE [Page 1] [Page 2] [Page x] [Table des matières] [Index]

La possibilité d'aller sur la page précédente ou suivante doit exister

NB: le caractère spécial **<** vaut < et **>** vaut >

- Dans une structure hiérarchique, on a une présentation plus complexe inspirée de celle d'un livre de plusieurs chapitres

Chaque tête de chapitre portant en haut un bandeau rappelant l'ensemble des chapitres et permettant d'y accéder directement,

Dans chaque chapitre (contenant plusieurs pages) l'utilisateur pourra revenir au début du chapitre précédant ou suivant et verra sur le haut de l'écran un bandeau rappelant l'ensemble des pages et permettant un déplacement direct sur toutes les pages du chapitre

Une table des matières référençant chapitres et pages sera également fournie voire si possible un index contenant des points d'entrée précis pour toutes les notions importantes.

Sur disque le stockage devrait se faire de façon à refléter la disposition du site.

IMAGES ET HTML

Il est important de pouvoir présenter des images sur notre site mais le problème des qualités et des tailles de fichiers est réel

Les deux formats existant sur le Web sont des formats compressés

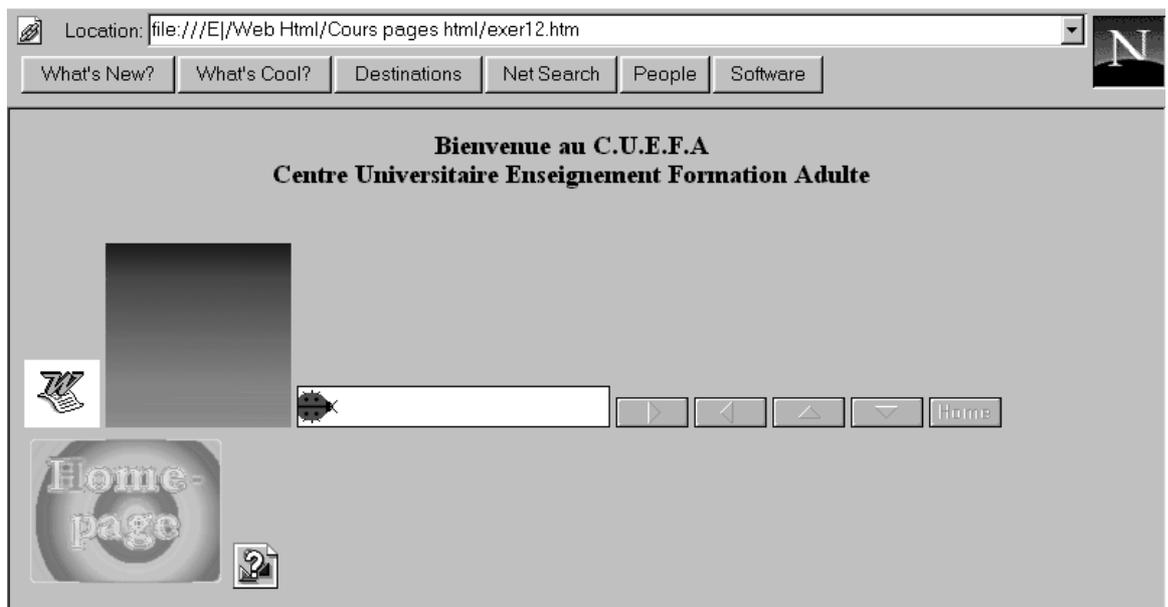
Les tags nécessaires sont les suivants

``

marqueur simple d'insertion de l'image référencée par le paramètre SRC="xxx" avec xxx le nom du fichier image voulu
N.B: la syntaxe ../dossier reste valable comme pour les liens

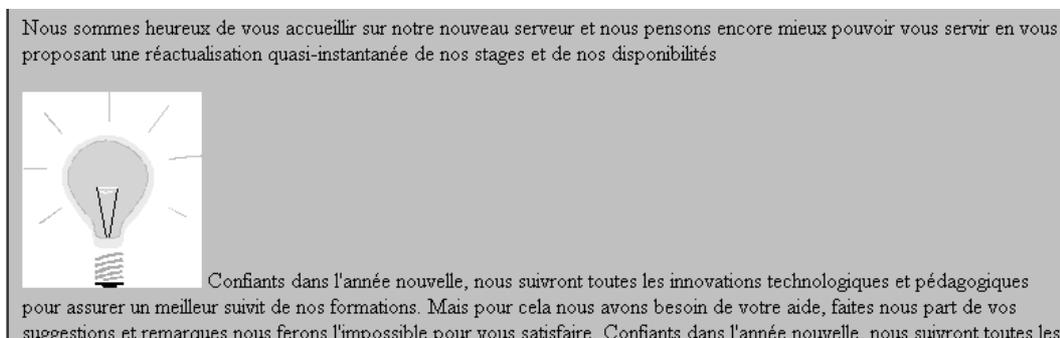
En cas de problème l'affichage du pictogramme  (sur Netscape) ou  (sur explorer) permettra de repérer l'incident

Ainsi le fichier **EXER12.HTM** affiche les images gif suivante



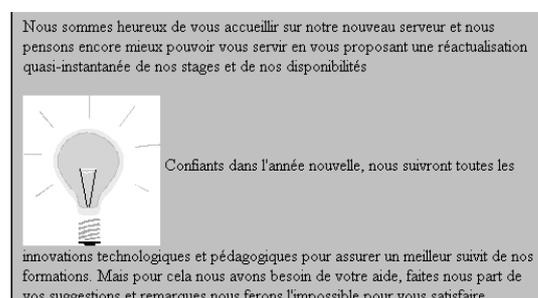
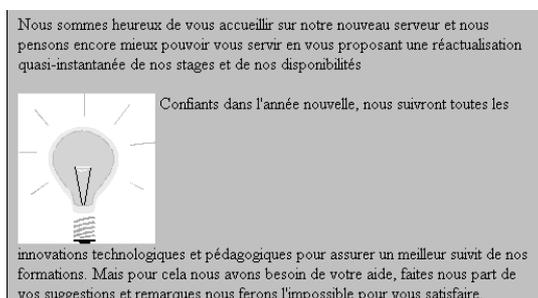
Positionner une image directement

Il est important de pouvoir placer précisément une image dans une page
par défaut le texte s'inscrit après l'image sur une seule ligne, puis la suite est positionnée "après" l'image"



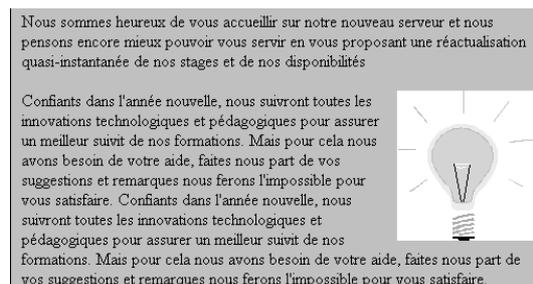
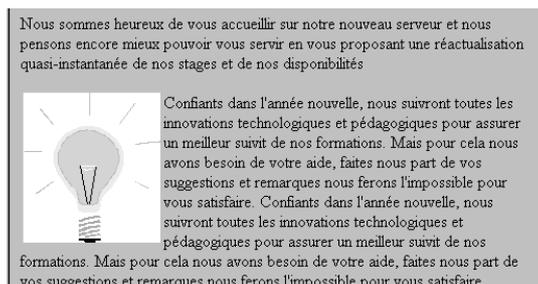
Le Tag `` accepte le paramètre défini par le mot clé `ALIGN="top"`, `"middle"` ou `"bottom"` permettant de placer un texte de légende sur le coté de l'image

N.B: une seule ligne est placée sur le coté de l'image, et si retour à la ligne il y a la suite du texte se placera après l'image !



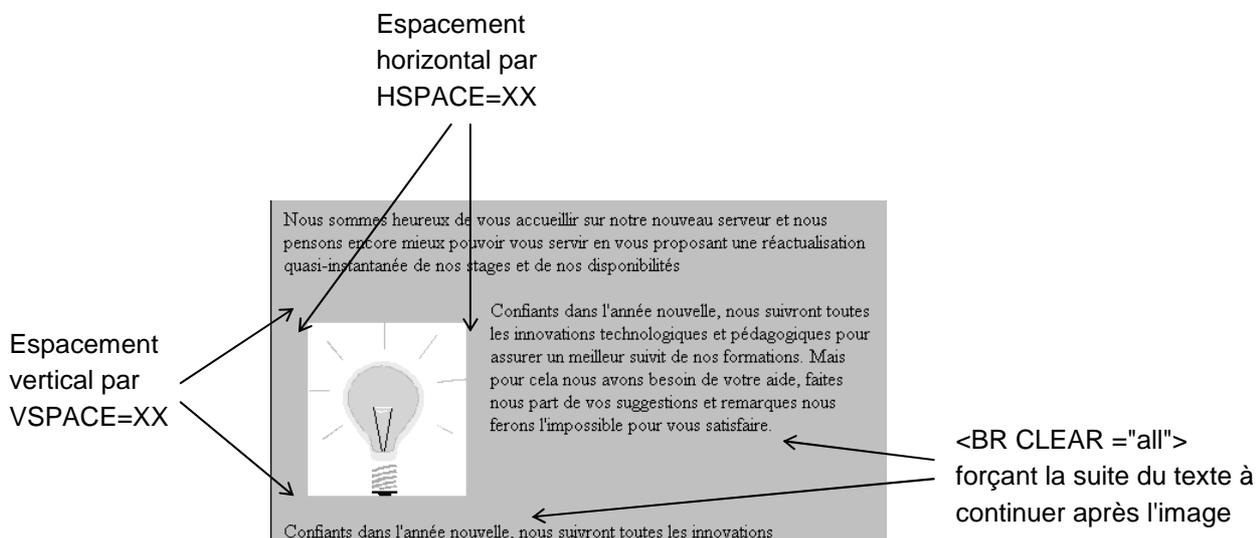
Le paramètre défini par le mot clé `ALIGN` accepte aussi les valeurs `ALIGN="left"`, ou `"right"` permettant de placer un texte de légende sur le coté gauche ou droit de l'image

N.B: plusieurs lignes peuvent être placées sur le coté de l'image, le texte "coulant" autour de celle-ci



Le Tag `` accepte le paramètre défini par le mot clé `VSPACE=xx` et `HSPACE=XX` avec `xx` en pixel définissant l'espacement à laisser avant ou après l'image

Enfin si on veut obtenir un retour à la ligne après l'image, de manière à ce que le texte suivant apparaisse à la suite, le tag `
` accepte le paramètre `CLEAR="all"`



(on trouvera le source dans **EXER13.HTM**)

Positionner une image dans un tableau

Il est parfois plus simple de positionner une image en créant un tableau, évidemment sans tracer sa bordure

Le principe reste celui de la construction classique de tableau avec simplement une dimension de la cellule contenant l'image exprimée en absolu en pixel

Ainsi si on incorpore une image, il vaut mieux connaître sa taille pour dimensionner correctement la cellule du tableau

En effet une image à une taille (hauteur et largeur) fixe exprimée en pixels, ne changeant pas selon la taille de la fenêtre du navigateur ou de la résolution de l'écran

par exemple le fichier "ampoule.gif" contient une image de 131x144 pixel en 256 couleurs

la cellule du tableau qui va la contenir devrait être prévue de la sorte

```
<TD WIDTH=131>
  <IMG SRC="xxxx/ampoule.gif">
</TD>
```

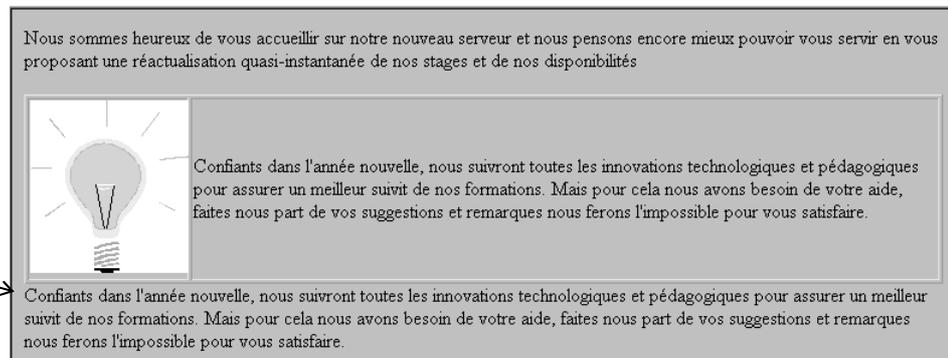
Définition de la largeur de cellule exactement semblable à la taille de l'image"

et la cellule à coté devrait occuper la totalité restante du tableau

```
<TD WIDTH="99%" ALIGN="Center">
  le texte que l'on veut voir s'afficher
  sur le coté de l'image
</TD>
```

Définition de la largeur de cellule pour remplir la totalité restante de la largeur du tableau
N.B : on peut aussi ne rien mettre

Le tableau n'est tracé ici que pour une meilleure visualisation
Le source se trouvant en **exer14.htm**



Taille et chargement d'une image

On peut modifier la taille de l'image à l'écran sans aucun problème, en incluant les dimensions en pixels de ce que l'on veut afficher

Le Tag `` accepte le paramètre défini par le mot clé `WIDTH=xx` et `HEIGHT=xx` avec `xx` en pixel définissant la taille de l'image

Ainsi si notre image précédente (`ampoule.gif`) qui mesurait à l'origine une taille de 131x144 pixels est affichée par le tag

```
<IMG SRC="xxx/ampoule.gif" WIDTH=262 HEIGHT=288>
```

elle se voit doubler de taille !! (chargez **exer15.htm** et commentez !)

Il apparaît donc utile de connaître les dimensions d'une image pour pouvoir l'afficher comme on le souhaite, mais il faut noter que **le temps de chargement dépend de la taille originale de l'image**, et que trois conséquences apparaissent:

- Il vaut mieux si possible ne pas demander d'afficher une image réduite, mais plutôt se créer une image de la taille voulue, ainsi le chargement sera plus rapide
- Il vaut mieux si possible ne pas demander d'afficher une image agrandie car un problème de proportions et de définition (qualité) se posera
- Dans le cas où l'on voudrait inclure dans une page deux fois la même image, il vaut mieux référencer deux fois la même image (et donc une réduite) car celle-ci ne sera chargée qu'une fois, alors que si on référence deux images (une grande et une petite) forcément le temps de chargement sera plus long

IMPORTANT

En général le temps de transmission d'une image est plus important que celui du texte, et par conséquent le navigateur reçoit d'abord le texte puis "attend" les images pour afficher le résultat, (pour composer la page)

On peut accélérer la vitesse d'affichage si on donne au browser la taille de l'image précisément, ainsi la mise en page pourra se faire et le texte s'affichera avant que l'image complète ne soit reçue

- Ainsi il vaut mieux toujours spécifier la taille de l'image à l'aide des paramètres `WEIDTH` et `HEIGHT` pour permettre l'affichage du texte en "attendant" la réception complète de l'image

image précédée d'une ... image

Dans le cas d'un chargement d'une image de grande taille (en octets) on peut demander de précharger une image plus simple puis après l'image réelle

Le Tag `` accepte le paramètre défini par le mot clé `LOWSRC="xxx"` et "xxx" étant les références de l'image à charger en premier

Ainsi dans l'écriture

```
...  
<IMG SRC="ampoule.gif" LOWSRC="attente.gif" WIDTH=131 HEIGHT=144>  
...
```

Il s'agit ici à terme de charger l'image AMPOULE.GIF

Cette image à une taille de 131x144 pixels

Et vu que le texte peut s'afficher immédiatement dans une bonne pagination charger d'abords l'image attente.gif (puis ampoule.GIF)

- **Attention : ne garder cette technique que pour les images globalement importantes car en définitive on rallonge le temps global de chargement de la page**

FICHIERS GRAPHIQUES

Bitmap-Vectoriels

Il existe différentes catégories de fichiers graphiques, les fichiers de type bitmap (ou raster), et les fichiers de type vectoriels (ou metafile)

- Le fichier de type **bitmap** peut être considéré comme une série de points de coordonnées x et y, chaque point ayant une intensité lumineuse et une couleur. le nombre de point total de l'image est appelé résolution

Avantage : c'est le format le plus rapide à l'affichage, le plus fidèle (les écrans aussi sont des périphériques par points) et le moins destructif lors de conversions,

Inconvénient : c'est le format qui peut être le plus volumineux pour une image précise et détaillée et c'est un format qui ne se prête pas aux agrandissements, voire même aux réductions

- Le fichier de type **vectoriel** implique l'idée selon laquelle on décrit l'image par un ensemble de segments de droite (traits) dont on ne stockerait pas la totalité des points mais seulement leurs extrémités et leurs couleurs

Avantages : Un fichier vectoriel est plus compacte qu'un fichier bitmap et sa taille ne varie pas avec la résolution du dessin, de plus les réductions et les agrandissements donnent d'excellents résultats

Inconvénient : c'est le format qui se prête le moins aux conversions

Tableau estimatif sur le temps de téléchargement d'une image de 400 ko

Vitesse de connexion (en bps)	Taux de transfert maxi	Temps nécessaire pour charger une image de 400 ko
9600	960 caract. / sec.	420 sec.
33600	3360 caract. / sec.	120 sec.
56000	5600 caract. / sec.	35 sec.
64000 (numéris)	6400 caract. / sec.	3,5 sec.
512 K (adsl bas débit)	512 000 caract. / sec.	0,5 sec.
5 Mg (adsl haut débit)	5 000 000 caract. / sec.	0,05 sec.

Résolution nécessaire

On peut avoir des images dont le nombre de couleur varie entre:

- 16 couleurs (mode vga classique)
- 256 couleurs (sur 8 bits)
- 65536 couleurs (sur 16 bits mode HiColor)
- 16 777 216 couleurs (sur 24 bits mode TrueColor)

Formats GIF, JPEG ...

Les fichiers utilisés dans les pages HTML sont des formats bitmaps compressés

xxxxxx.GIF : format permettant de gérer des images couleurs au format GIF (Graphic Interchange Format) en 16 ou 256 teintes voire plus par palettes interposées
Créé par CompuServe en 1987

xxxxxx.JPG : format permettant de gérer des images couleurs au format JPEG (Joint Photographic Expert Group)

xxxxxx.PNG : Format nouveau crée par le W3C en octobre 1996
PNG (portable Network Graphics)
Si ce format est réellement intéressant techniquement, il ne semble pas être adopté dans la pratique par énormément de personnes

Tableau comparatif des 3 formats utilisables sur les pages HTML:

	GIF	JPEG	PNG
Compatibilité	Palette 8 bits	RVB Niveaux de gris	RVB Niveaux de gris Palette 8 bits
Compression	sans perte de 4:1 à 10:1	Avec perte de 10:1 à 100:1	Sans perte et de 5% à 25% meilleur que GIF
Transparence	2 niveaux par pixel (gif89a)	Non	254 niveaux par pixel
Affichage progressif (image entrelacée)	Oui 1/8 d'image	Oui Progressive JPEG	Oui 1/64 d'image
Animation	Oui	Non	Non*

Formats images principaux

Bitmap ↔	All Files	
	BMP - OS/2 or Windows Bitmap	
	CDR - CorelDraw Drawing	
Vectoriel ↔	CGM - Computer Graphics Metafile	
	CLP - Windows Clipboard	
	CUT - Dr. Halo	
	DIB - OS/2 or Windows DIB	
	DRW - Micrografix Draw	
	DXF - Autodesk Drawing Interchange	
	GEM - Ventura/GEM Drawing	
	GIF - CompuServe	GIF un des formats supportés en natif sur le WEB
Bitmap ↔	HGL - HP Graphics Language	
	IFF - Amiga	
Bitmap ↔	IMG - GEM Paint	
Bitmap ↔	JIF - JPEG - JFIF Compliant	
Bitmap ↔	JPG - JPEG - JFIF Compliant	
	LBM - Deluxe Paint	
	MAC - MacPaint	
	MSP - Microsoft Paint	
	PBM - Portable Bitmap	
Bitmap ↔	PCD - Kodak Photo CD	
	PCT - Macintosh PICT	Indétrônable sur Macintosh
Bitmap ↔	PCX - Zsoft Paintbrush	
	PGM - Portable Graymap	
	PIC - PC Paint	
	PIC - Lotus PIC	
	PNG - Portable Network Graphics	Référence en ce qui concerne la retouche d'images
	PPM - Portable Pixelmap	
	PSD - Photoshop	
	RAS - Sun Raster Images	
	RAW - Raw File Format	
	RLE - CompuServe or Windows	
Bitmap ↔	TGA - Truevision Targa	
Vectoriel ↔	TIF - Tagged Image File Format	Indétrônable sur Macintosh
	WMF - Windows Meta File	
	WPG - WordPerfect Bitmap	
	WPG - WordPerfect Vector	

IMAGES ET LIENS

A partir du moment où une image est incorporée, on peut la transformer en un lien hyper texte

Pour créer un hyper lien à partir d'une image il suffit d'insérer son tag IMG là où auparavant on insérait du texte

```
<A HREF="typelien"> <IMG SRC="ampoule.gif" WIDTH=131 HEIGHT=144> </A>
```

Cette image devenant un lien elle se trouve encadrée mais on peut enlever cette bordure par le paramètre BORDER=0 dans le tag

reprenre les 3 pages **EXER10_1.HTM**, **EXER10_2.HTM**, **EXER10_3.HTM** et remplacer la mention SUIVANT et PRECEDANT en bas des pages par les images fournies GAUCHE.GIF et DROITE.GIF

(correction en **EXER16_1.HTM**, **EXER16_2.HTM** et **16_3.HTM**)

Bordure bleue indiquant un lien (annulable par BORDER=0)

Pointeur de souris en forme de main (pointeur sur un lien image)

Adresse de la page visitée

Image DROITE.GIF placée en hyper lien vers la page 2

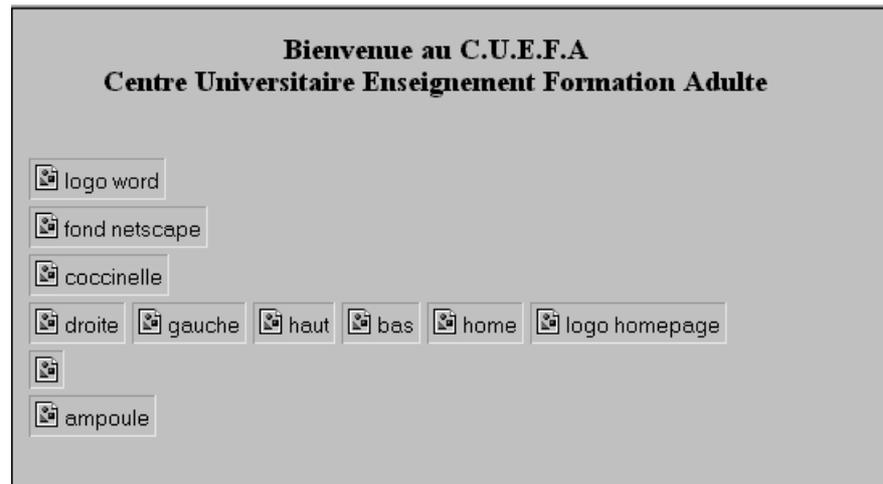
Adresse de la page pointée dite URL

Nommer des images

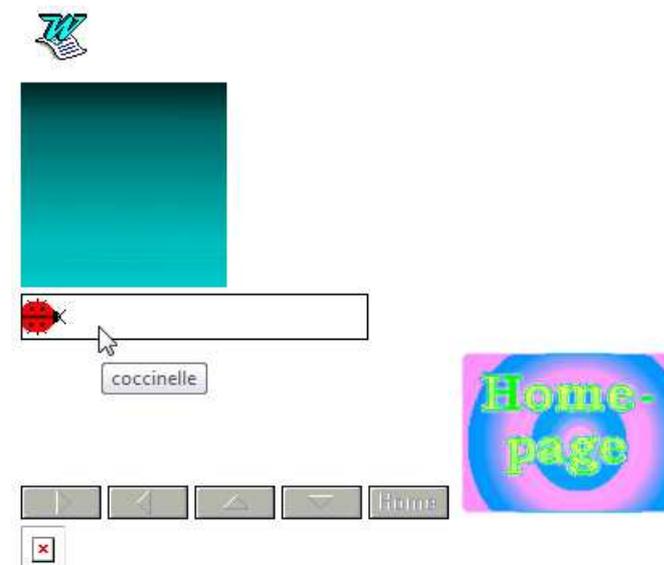
Il est intéressant de nommer une image pour deux raisons, l'utilisateur peut avoir un navigateur de type texte, ou bien simplement avoir désactivé l'affichage systématique de toutes les images pour économiser les temps de chargement

Utiliser pour cela le paramètre ALT="texte" dans le tag

Dans ce cas l'affichage des images se fera par le biais d'un pictogramme et seul le texte mentionné par le paramètre ALT est affiché (cf **EXER17.HTM**)



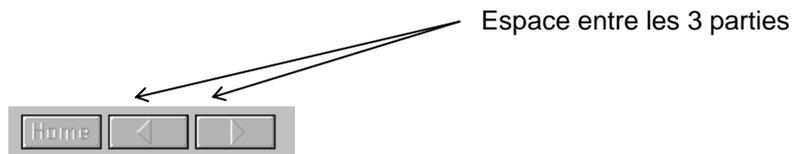
Remarque : dans le cas où les images s'affichent, cette nomination via le paramètre ALT du tag permet de faire afficher une bulle d'aide lorsque l'utilisateur passe la souris au dessus de l'image.



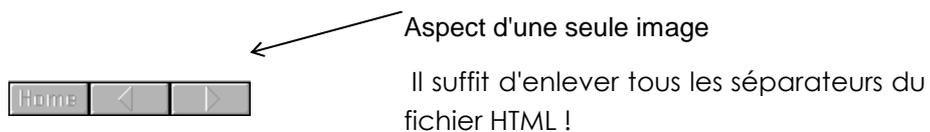
Combiner des images

On peut combiner des images entre elles pour donner à l'utilisateur l'impression qu'il s'agit d'une seule image, et ainsi donner à chaque partie de l'image un lien précis

Essayez de créer "une seule image" à partir des fichiers **Home.gif**, **Gauche.gif** et **Droite.GIF**



```
<IMG SRC="images/gif/home.gif" ALT="home">  
<IMG SRC="images/gif/gauche.gif" ALT="gauche">  
<IMG SRC="images/gif/droite.gif" ALT="droite">
```



```
<IMG SRC="images/gif/home.gif" ALT="home"><IMG  
SRC="images/gif/gauche.gif" ALT="gauche"><IMG  
SRC="images/gif/droite.gif" ALT="droite">
```

PARAMETRES DE PAGE

Couleur - arrière plan ...

Il est possible de demander de changer le fond d'écran de votre page HTML

Les tags nécessaires sont les suivants

`<BODY BGCOLOR= "x">`

marqueur simple définissant la couleur de fond (background) de la page HTML
Le navigateur adapte toutes les autres couleurs par défaut en fonction (liens) mais pas le texte

N.B: la syntaxe "x" en hexadécimal donne "#000000" reste valable comme pour le Texte
Dans cette valeur on a le Rouge, le Vert et le Bleu pouvant chacun avoir une palette de 256 nuances

Ainsi ce codage en hexadécimal permet de "lire" la teneur respectivement en Rouge Vert et Bleu du texte avec 00 mini et FF maxi (255).

Par exemple : 000000 est noir FF0000 est rouge

00FF00 est vert 0000FF est bleu

FFFFFF est blanc

Chargez le fichier **COULEUR.HTM** dans le navigateur

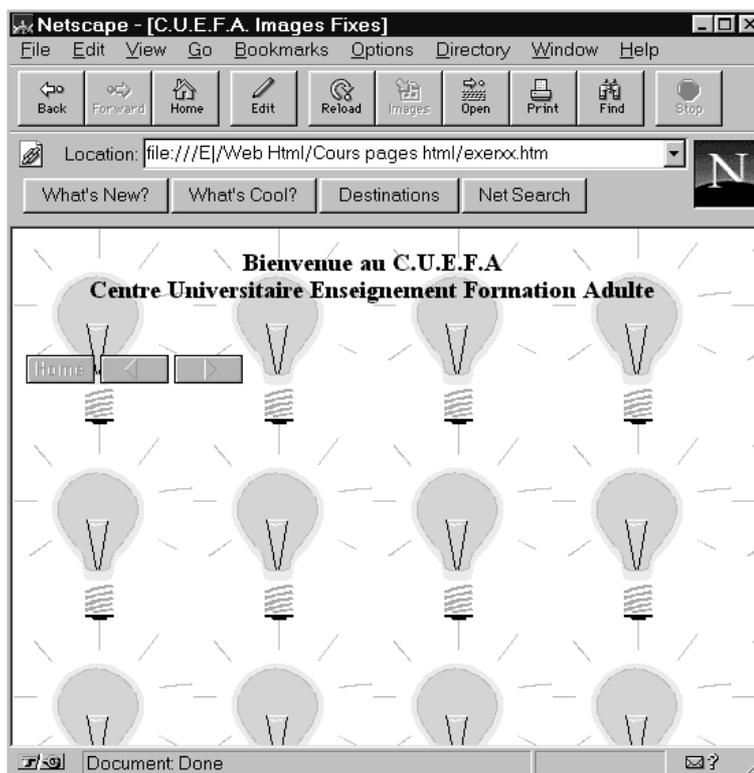
Essayez de charger le fichier **EXER18.HTM** avec un fond noir, puis jaune...

Solution en **EXER19.HTM**. Regarder le fichier **PRESENTE.HTM**

N.B: Par conséquent pour assurer la lisibilité d'un texte, même dans le cas où un navigateur ne comprendrait pas un tag couleur, il faut demander un fond clair et une couleur de texte foncée (ou l'inverse)



Obtenir cet effet repose sur la possibilité de charger une image en fonds d'écran
(EXER20.HTM)



<BODY BACKGROUND= "xx">

marqueur simple définissant l'image de fond "xx"(background) de la page HTML. Le navigateur utilisera cette image comme un "carreau" et la répètera de façon à couvrir la totalité de l'écran.

Liens hyper-texte

On s'aperçoit que le tag BODY peut recevoir différents paramètres, résumons les

<BODY LINK= "x">

LINK="#000000" couleur des liens

<BODY ALINK= "x">

ALINK="#000000" couleur lien sélectionné

<BODY VLINK= "x">

VLINK="#000000" couleur de lien visité

FORMULAIRES (CRÉATION)

Beaucoup de pages contiennent des formulaires qui, une fois saisis par l'utilisateur, peuvent être transmis au serveur WEB hébergeant le site

Pour préparer un formulaire il faut une zone d'édition (appelée FORM), puis définir la méthode à employer pour transmettre au serveur l'information recueillie dans les champs du formulaire

Dans un formulaire, tous les tags de mise en forme sont disponibles classiquement, Il faut simplement éviter d'incorporer un formulaire dans un autre formulaire ou de croiser un formulaire et un tableau

Visualisez le formulaire du fichier **EXER22.HTM**

NOM:

Zone de texte ou Text Box,

Zone masquée ou Password Box,

Mr Mdme Mlle

Boutons Radio ou Radio Button

Français
 Anglais
 Italien
 Allemand

Cases à Cocher ou Check Box

Listes Déroulantes ou Select input

Scrolling text

ou

Zone de texte

Structure Générale

Les tags nécessaires sont les suivants

<code><FORM></code>	<code></FORM></code>	marqueur de début et de fin de formulaire
deux paramètres principaux existent pour un formulaire		
<code>METHOD="type"</code>		indiquant quel protocole il faut utiliser pour envoyer les données
		"Post" est le protocole à utiliser en général "get" étant l'alternative
<code>ACTION="adresse"</code>		indiquant l'endroit où les résultats de la saisie dans le formulaire doivent être envoyés pour être traités, sous forme généralement d'une URL de @mail

Ainsi notre formulaire s'inscrira le plus souvent dans la trame suivante

```
<FORM METHOD="post" ACTION="adresse URL de l'exécutable">  
  
  Corps du formulaire  
  
</FORM>
```

Ensuite le corps du formulaire est composé de 3 tags principaux

INPUT
SELECT
TEXTAREA

Chacun de ces tags amenant une zone de saisie à l'écran, se voit assigner d'un paramètre `NAME="nom"` avec "nom" étant un identificateur donné pour la valeur saisie par l'utilisateur dans le formulaire à ce niveau (et donc pour pouvoir ensuite s'en servir ultérieurement)

Souvent ces tags acceptent un paramètre `VALUE="x"` permettant d'indiquer le plus souvent la valeur par défaut à renvoyer

Enfin chacun de ces tags accepte des paramètres biens spécifiques selon la zone de saisie qu'il crée

le tag INPUT

C'est le tag qui va permettre les entrées saisies au clavier principalement.

Sa structure générale étant :

<INPUT TYPE="xx"> avec "xx" indiquant de quelle façon les données seront saisies par l'utilisateur
xx valant "text", "password", "radio", "checkbox"

TYPE="text" permet de créer un champs de saisie, c'est le type par défaut si rien n'est spécifié

```
<INPUT TYPE="text" Name="n" SIZE=25  
MAXLENGTH=25 VALUE="tapez votre nom ici">
```

génère  la valeur saisie ira dans une variable nommée "n". C'est le type par défaut

TYPE="password" idem "text" mais l'utilisateur ne voit pas les lettres tapées au clavier

génère



TYPE="radio" permet de créer des "boutons radio", un seul bouton est sélectionnable sur l'ensemble des possibilités

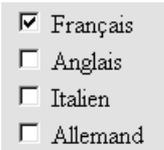
```
<INPUT TYPE="Radio" NAME="q" VALUE=1 CHECKED>Mr  
<INPUT TYPE="Radio" NAME="q" VALUE=2>Mme  
<INPUT TYPE="Radio" NAME="q" VALUE=3>Mlle
```

génère  et la valeur saisie (1,2 ou 3) ira dans une variable nommée "q" (pour qualité...)

Le paramètre CHECKED active un bouton par défaut

TYPE="checkbox" permet de créer des "cases à cocher", plusieurs cases sont sélectionnables dans l'ensemble

```
<INPUT TYPE="Checkbox" NAME="langues" VALUE="Fr"  
CHECKED>Français<BR>  
<INPUT TYPE="Checkbox" NAME="l" VALUE="Gb">Anglais <BR>  
<INPUT TYPE="Checkbox" NAME="l" VALUE="Ita">Italien <BR>  
<INPUT TYPE="Checkbox" NAME="l" VALUE="All">Allemand<BR>
```

génère  et la valeur saisie (Fr, Gb, Ita ou All) ira dans une variable nommée "l" (pour langue)...

Le paramètre CHECKED active une case par défaut

le tag SELECT

C'est le tag qui va permettre des entrées sélectionnées principalement

Sa structure générale étant

`<SELECT>` `</SELECT>` encadrant l'ensemble des valeurs constituant la liste, Chacune de ces valeurs étant précédée du tag `<OPTION>`. Si `MULTIPLE` est précisé une sélection multiple via la touche `CTRL` est autorisée

```
<SELECT NAME="h" SIZE=4>
<OPTION>08h00<OPTION>08h30<OPTION SELECTED>09h00
<OPTION>09h30<OPTION>10h00<OPTION>10h30<OPTION>11h00
<OPTION>11h30<OPTION>12h00<OPTION>13h30<OPTION>14h00
<OPTION>14h30<OPTION>15h00<OPTION>15h30<OPTION>16h00
</SELECT>
```



Génère et la valeur sélectionnée ira dans une variable nommée "h" (pour heure...)

Le paramètre `SELECTED` sélectionne une valeur par défaut

le tag TEXTAREA

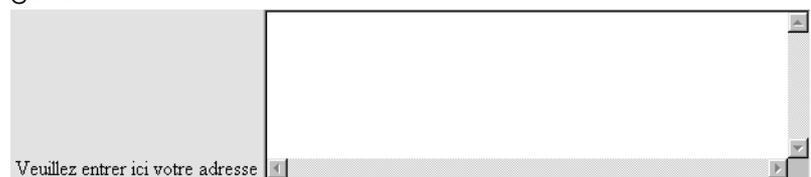
C'est le tag qui va permettre des entrées clavier volumineuses

Sa structure générale étant

`<TEXTAREA>` `</TEXTAREA>` Permet de définir dans un formulaire extensible "à volonté" par l'utilisateur une zone de texte lors de sa saisie Si `ROWS` et `COLS` sont précisés une taille en lignes et colonnes sera donnée

```
Veillez entrer ici votre adresse
<TEXTAREA NAME="ad" ROWS=5 COLS=40> </TEXTAREA>
<BR>
```

génère



et la valeur saisie ira dans une variable nommée "ad" (pour adresse...)

Annulation ou Envoi

La fin d'un formulaire est toujours constitué de deux boutons permettant soit de remettre à zéro les saisies effectuées dans le formulaire (abandon) soit d'envoyer le formulaire

Leur structure étant

`<INPUT TYPE="submit" VALUE="xxx">` Pour le bouton qui déclenchera l'envoi des valeurs saisies dans le formulaire "xxx" étant le libellé du bouton

```
<INPUT TYPE="submit" VALUE="Envoi">
```

génère  et les valeurs saisies seront expédiées

`<INPUT TYPE="reset" VALUE="xxx">` Pour le bouton qui effacera toutes les valeurs saisies dans le formulaire "xxx" étant le libellé du bouton

```
<INPUT TYPE="reset" VALUE="Annuler">
```

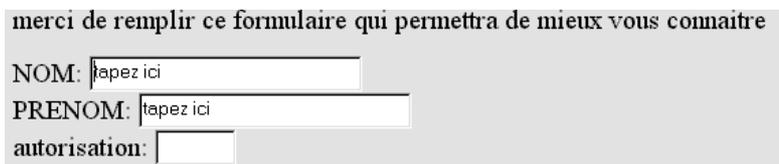
génère  et les valeurs saisies seront effacées

Présentation

Les formulaires permettent de saisir énormément de choses, mais demandent une présentation soignée

Les listes et les tableaux restent à disposition mais il faut faire attention à respecter la syntaxe des tags

Ainsi à la place de



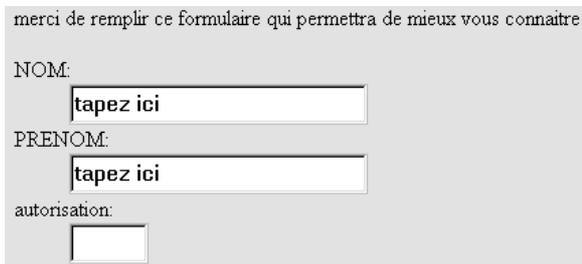
merci de remplir ce formulaire qui permettra de mieux vous connaître

NOM:

PRENOM:

autorisation:

on préférera



merci de remplir ce formulaire qui permettra de mieux vous connaître

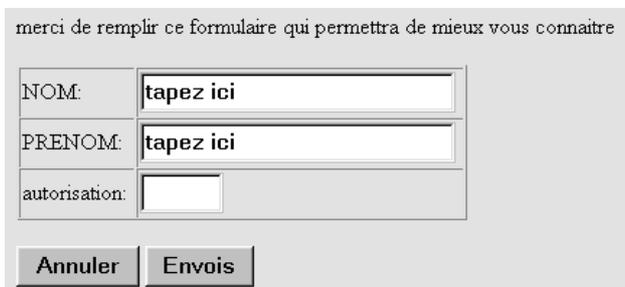
NOM:

PRENOM:

autorisation:

fondé sur des liste comme dans **EXER23.HTM**

ou bien aussi



merci de remplir ce formulaire qui permettra de mieux vous connaître

NOM:	<input type="text" value="tapez ici"/>
PRENOM:	<input type="text" value="tapez ici"/>
autorisation:	<input type="text"/>

fondé sur un tableau comme dans **EXER24.HTM**

FORMULAIRES (TRAITEMENT)

principes

Les formulaires permettent de saisir énormément de choses, mais demandent un traitement sortant du cadre HTML proprement dit, dans ce que l'on appelle des scripts

Un script est un programme acceptant certaines données saisies par le visiteur de la page, les traitants et proposant ensuite éventuellement un résultat

Cela peut consister simplement à rajouter un nom à une liste de publipostage comme à effectuer des calculs beaucoup plus complexes avec des données fournies par l'utilisateur

HTML ne supporte pas en lui-même ce genre de fonctionnalités car c'est un langage de description de page et non pas de traitement d'information, mais il peut faire appel à un script résidant sur le client ou le serveur, étant capable de lui passer des informations et d'en recevoir...

Il existe aujourd'hui plusieurs méthodes d'écriture de scripts de fonctions interactives dans les pages WEB, des exemples connus étant

Les scripts **CGI** (Common Gateway Interface) : la plus ancienne

Les scripts **PHP** : tendant à se répandre

Les règles d'écriture dépendent donc de l'environnement et du serveur sur lequel les traitements doivent être exécutés, les serveurs UNIX acceptant très bien les scripts CGI, les serveurs WINDOWS acceptant d'autres langages, tels que ASP.

formulaire via "mailto"

On a dit que notre formulaire s'inscrira le plus souvent dans la trame suivante

```
<FORM METHOD="post" ACTION="adresse URL de l'exécutable">
```

Corps du formulaire

```
</FORM>
```



L'action peut utiliser un lien "mailto"... comme dans

```
<form action="mailto:michel@cabare.net?subject=formulaire" method="post"
name="inscription" ENCTYPE="text/plain">
```

dans ce cas, l'envoi réussit depuis une machine ayant un client mail correctement paramétré, de ce formulaire

site de démonstration

particulier	toto
Ville	grenoble
Adresse Email	toto@pp
Téléphone	04.76.26.77.25
Adresse postale (si envois par courrier)	espoir

obtenu via **formulaire-mailto.htm**

```
<p>site de démonstration</p>
<form action="mailto:michel@cabare.net?subject=formulaire" method="post" name="inscription" ENCTYPE="text/plain">
  <table border="0">
    <tr>
      <td><select name="genre" size="1" id="genre">
        <option >société; entreprise</option>
        <option selected="selected">structure</option>
        <option>particulier</option>
      </select></td>
      <td><input name="vgenre" type="text" size="25" maxlength="40" /></td>
    </tr>
    <tr>
      <td>ville</td>
      <td><input name="vville" type="text" size="25" maxlength="40" /></td>
    </tr>
    <tr>
      <td>Adresse Email</td>
      <td><input name="vemail" type="text" size="25" maxlength="40" /></td>
    </tr>
    <tr>
      <td>Téléphone</td>
      <td><input name="vtel" type="text" size="14" maxlength="20" /></td>
    </tr>
    <tr>
      <td colspan="2">Adresse postale (si envois par courrier)
      <textarea name="vadresse" cols="35" rows="6" ></textarea></td>
    </tr>
    <tr align="center">
      <td><input name="Annuler" type="reset" id="Annuler" value="Annuler" /></td>
      <td><input name="Envoyer" type="submit" id="Envoyer" value="Envoyer" /></td>
    </tr>
  </table>
</form>
```

donnera au destinataire le mail suivant :

```
De : cabaré michel [michel.cabare@wanadoo.fr]
À : michel@cabare.net
Cc :
Objet : formulaire

|
genre=particulier
vgenre=toto
vville=grenoble
vemail=toto@pp
vtel=04.76.26.77.25
vadresse=espoir
Envoyer=Envoyer
```

POLICE PROPORTIONNELLE OU NON

A partir du moment où le navigateur effectue lui-même les sauts de ligne et se charge des espaces, on aura du mal à présenter une page HTML en "alignant" des motifs par des espaces ou des retours à la ligne

Ainsi pour les listing informatiques, ou les petits logo créés à l'aide des codes ASCII, les présentations en tableau se posent un problème certain, si on veut forcer le navigateur à respecter la présentation du source HTML, au caractère près, y compris les espaces et les retours à la ligne il faut utiliser les tags

`<PRE>` `</PRE>` écris avec une police non proportionnelle et respecte les espaces et les sauts de ligne effectués dans l'éditeur

essayer de "passer" ce texte dans un fichier HTML sans l'encadrer des tags `<PRE>` et `</PRE>`, le résultat est impressionnant n'est ce pas ?

Nom	votes obtenus	%
<hr/>		
candidat 1:	670 votes	59
candidat 1:	341 votes	30
candidat 1:	123 votes	11
<hr/>		
total des votes :	1134 votes	100

ÉDITEURS & GENERATEURS

Le monde du WEB est un monde en pleine mouvance, et donner ou tenter une liste d'éditeurs ou d'utilitaires serait utopique

On fera simplement quelques distinctions de catégories importantes

Editeurs HTML

- Il existe des éditeurs permettant au minimum de ne pas avoir à taper les tags à la main mais les générant via des boutons ou des outils que l'utilisateur sélectionne
- il existe des éditeurs permettant de masquer carrément la page HTML pour visualiser directement les effets construits, on parle alors d'éditeur "WYSIWYG"

Gestionnaires de Site

- En plus d'un éditeur, un gestionnaire de site doit impérativement intégrer un mécanisme de gestion de liens

Conversion en HTML

- Logiciels de conversion de format texte en HTML "moulinettes" de transformation de fichiers RTF en HTML
- Outils bureautiques, au sens large du terme, permettant de générer une page html plus ou moins "propre"



TEMPS DE ... LECTURE

Sans revenir sur les conseils pour diminuer la taille de vos fichiers, comment avoir une idée du temps de chargement de vos pages pour vos lecteurs ?

Deux paramètres interviennent, la taille de vos pages et la vitesse de connexion de votre lecteur (sans parler du débit réel de votre serveur ou de son prestataire d'accès...)

Taille des pages

Un moyen simple donnant une idée de la taille de vos pages consiste à regarder dans l'explorateur, après avoir sélectionné vos fichiers, la taille de la sélection

Vitesse de chargement

Quelques vitesses existent :

33000-56000 (4125-7000 octets/s) modem analogiques,

4000-128000 b/s correspondant à du numeris ou ADSL 1^o niveau (8000-16000 octets/s)

soit environ 1800, 3500, 4125, ou 8000 octets par seconde

Par conséquent si votre page complète fait :

- de 1 à 20 Ko :
rapidement transmise même avec des vitesses lentes
- de 20 à 50 Ko :
peut mettre jusqu'à 1 minute pour être chargée, et une minute à attendre devant un écran, c'est long !
- au delà de 50 Ko :
Il vaudrait mieux fractionner la page en plusieurs pages plus réduites sous peine de voir votre site déserté !



COMMENT STRUCTURER UN SITE WEB

Quoi mettre sur le site Web

Un site Web, pour quoi faire ? Voilà bien la question essentielle qu'il convient de se poser avant de se lancer dans sa réalisation. Au-delà de l'attrait initial ou de l'effet de mode, la décision d'utiliser ce nouveau média de communication mérite en effet qu'on s'y attarde un certain temps.

La meilleure manière de répondre à cette question est de se demander à tout moment ce que le lecteur devra avoir retenu lorsqu'il aura terminé la lecture de chacune des pages du site. En effet, l'objectif recherché lors de la réalisation d'une page personnelle est bien différent de l'objectif d'une société de vente par correspondance qui souhaite réaliser un catalogue en ligne. D'une manière générale, plus les motivations pouvant attirer les visiteurs vers le site seront nettes, mieux ce sera. Et comme la seule limitation du Web est l'imagination du concepteur, il ne faut pas hésiter à se surpasser pour attirer et fidéliser le visiteur.

Comment structurer un site Web

Une fois que le but et les objectifs ont été clairement définis et que la matière première (textes, images, sons...) a été rassemblée, il convient de mettre de l'ordre dans ces différentes données en les regroupant par sujets dans des rubriques distinctes. Après ce premier découpage, il est indispensable de synthétiser le contenu de chaque rubrique pour ne retenir que l'essentiel qui sera finalement publié.

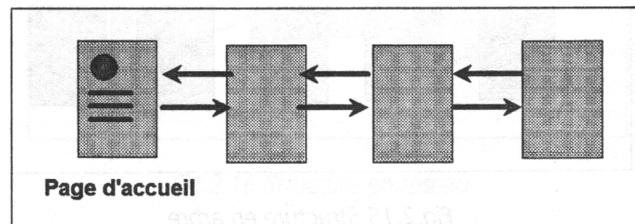
L'étape suivante consiste à placer ces différentes rubriques dans une structure hiérarchique en fonction des objectifs fixés lors de la définition du site.

A ce sujet, il existe quelques types de structures standard qui sont habituellement utilisés:

- structure linéaire
- structure en arbre
- structure en réseau
- structure mixte

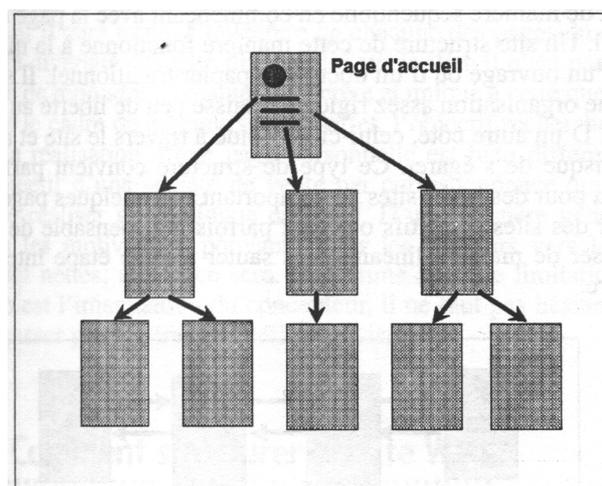
Structure linéaire

Dans une telle structure les différentes pages se suivent de manière séquentielle en commençant avec la page d'accueil. Un site structuré de cette manière fonctionne à la manière d'un ouvrage ou d'un document papier traditionnel. Il s'agit d'une organisation assez rigide qui laisse peu de liberté au visiteur. D'un autre côté, celui-ci est guidé à travers le site et a peu de risque de s'égarer. Ce type de structure convient parfaitement pour des petits sites ne comportant que quelques pages ou pour des sites éducatifs où il est parfois indispensable de progresser de manière linéaire sans sauter aucune étape intermédiaire.



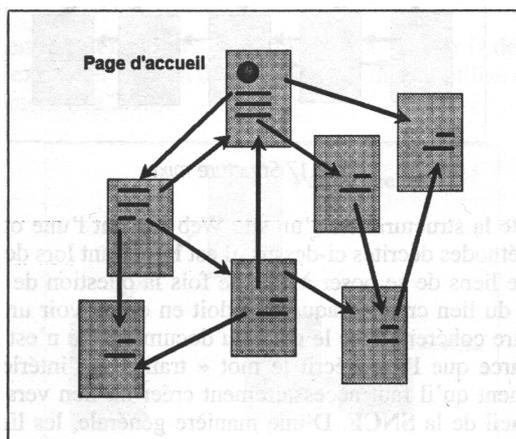
Structure en arbre

Dans une telle structure les différentes pages sont organisées de manière hiérarchique comme les branches d'un arbre. Le visiteur peut choisir de suivre la branche qui l'intéresse, ce qui l'amène à un « embranchement » où il peut à nouveau s'orienter dans la direction souhaitée. Une structure de ce type oblige le visiteur qui souhaite naviguer d'une branche horizontale à une autre, de remonter l'arborescence jusqu'au point de départ. Ce principe a l'avantage de ne pas désorienter le visiteur. Il est en effet facile pour celui-ci de se situer dans la structure du site et de s'y déplacer vers le haut s'il souhaite une information plus générale, ou vers le bas s'il souhaite avoir plus de détails. Pour ne pas lasser le lecteur il est conseillé de ne pas créer plus de deux ou trois niveaux.



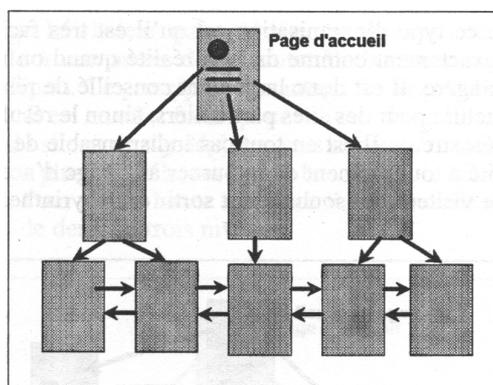
Structure en réseau

Dans une telle structure les différentes pages sont organisées sous la forme d'une toile d'araignée sans structure clairement définie. Le visiteur se déplace dans un site de ce type uniquement en suivant les liens qu'il rencontre au hasard des pages. Une telle structure convient parfaitement pour des sites comportant des jeux d'aventures ou des explorations virtuelles de lieux particuliers comme un musée ou une ville. Le problème avec ce type d'organisation est qu'il est très facile de s'y égarer, exactement comme dans la réalité quand on visite une ville étrangère. Il est donc largement conseillé de réserver une telle structure pour des sites particuliers, sinon le résultat risque d'être désastreux. Il est en tout cas indispensable de donner la possibilité à tout moment de retourner à la page d'accueil pour libérer le visiteur...



Structure mixte

Une structure mixte regroupe habituellement les caractéristiques des structures linéaires et en arbre. Ainsi le haut de la structure est de type arbre et le dernier niveau de type linéaire. Elle peut être utile dans un site où toute l'information détaillée se trouve au dernier niveau et que l'on souhaite la parcourir en vitesse sans passer par une structure hiérarchique. Pour ne pas désorienter le visiteur, il est utile de prévoir sur chacune des pages, des liens permettant de retourner en arrière, d'aller vers l'avant, vers le haut



La structure des pages Web

L'étape de structuration d'un site Web doit être complétée par la structuration de chacune de ses pages. Il faut à ce niveau faire la distinction entre la page d'accueil du site et les autres pages.

La page d'accueil

Quelle que soit la manière dont un site Web est structuré et quel que soit le nombre de pages qu'il contient, le point d'entrée est désigné sous le nom de «page d'accueil» (Home Page). Bien que les visiteurs aient la possibilité d'accéder directement à n'importe quelle autre page du site, via l'utilisation des outils de recherche par exemple, l'adresse de la page d'accueil correspond à l'adresse générale du site.

La page d'accueil doit être conçue de manière à :

- présenter l'objet du site;
- présenter une vue générale du contenu qui suit;
- établir les liens principaux entre les pages situées plus bas dans la hiérarchie;
- présenter éventuellement quelques liens vers d'autres sites;
- indiquer l'adresse électronique du responsable du site.

La page d'accueil étant la page principale du site, elle doit donc impressionner le visiteur, générer de l'intérêt et de la curiosité pour amener celui-ci à continuer l'exploration du site.

Les pages secondaires

Les pages secondaires constituent le corps du site. Comme le visiteur peut arriver sur chacune des pages secondaires à partir de n'importe quel autre endroit du Web, il est essentiel que chaque page se suffise à elle-même afin de ne pas désorienter celui-ci. D'autre part, le visiteur doit également pouvoir naviguer à travers le site, à partir de la page dans laquelle il vient d'arriver à pieds joints. Il est donc important de l'informer sur sa position par rapport à la hiérarchie du site et de lui donner la possibilité de retourner à tout moment à la page d'accueil.

La réalisation des pages secondaires nécessite ainsi les conseils suivants:

- découper l'information de la page à l'aide de titres et de sous-titres pour aérer le texte;
- établir un lien vers la page d'accueil à partir de chacune des pages secondaires;
- traiter un sujet par page;
- ne pas éclater un même sujet sur plusieurs pages;
- susciter la curiosité du lecteur par une page « événements »...

LES TAGS HTML (EN LISTE)

Les tags peuvent être écrits soit en minuscule, soit en majuscule, mais les conventions préfèrent une notation en capitale. Ils sont toujours inscrits entre signes de la sorte <tag>

On distingue deux types de tags, les tags qui ont besoin d'être désactivés et ceux qui n'ont pas besoin d'être désactivés.

Lorsqu'ils activent une situation, les tags prennent effet par une commande de type <tag> et par conséquent doivent désactiver la situation lorsqu'elle n'est plus souhaitée par une commande type </tag>. Lorsqu'ils ont une action ponctuelle, ils s'écrivent simplement <tag>

<!--	-->	marqueur de début et de fin de commentaire
------	-----	--

 xxx		marqueur début / fin de lien
--------------------------	------	------------------------------

Ce tag nécessite que le paramètre "typelien" définisse quel lien il va falloir mettre en œuvre, en respectant les notations suivantes:

HREF="C:\SITE\MONFICH.HTM"	un fichier HTM en local
HREF="http://xxx"	l'adresse xxx de sites sur le WEB
HREF="ftp://xxx"	adresse de dossiers ou fichiers sur un serveur ftp
HREF="telnet://xxx"	adressage de serveurs telnet
HREF="gopher://xxx"	adressage de dossiers ou de fichiers sur un serveur gopher
HREF="newsxxx"	adressage de Newsgroup
HREF="mailto:xxx"	Adressage d'adresse @Mail

Ce tag nécessite que la valeur de xxx définisse le texte qui renverra au lien lorsque l'on cliquera dessus

	marqueur simple d'insertion de son référencé par le paramètre HREF="xxx" avec xxx le nom du fichier son désiré
	 marque de repère Ce tag cré un repère "nom" référencable par une marque de type <A HREF.....> incluant la mention #"nom"
<BODY> </BODY>	marqueur de début et de fin de corps d'une page modifiable par BGCOLOR="#000000" couleur de page BACKGROUND="xx" image de fond page LINK="#000000" coul. des liens ALINK="#000000" coul. de lien sélectionné VLINK="#000000" coul. de lien visité
 	met en gras (Bold) peut s'insérer même autour d'un caractère dans un mot
<BLINK> </BLINK>	défini la portion de texte clignotante
 	provoque un saut de ligne Modifiable par CLEAR="all"
<CAPTION> </CAPTION>	marqueur de début et de fin de présentation de tableau, c'est à dire d'un titre hors tableau mais parfaitement centré sur lui. Doit être placé entre <Table> et la première rangée Modifiable par ALIGN="top" "bottom"

<DL>	</DL>	<p>marqueur de début et de fin de liste un saut de ligne avec ligne blanche est inséré avant et après encadre généralement des tag DT et DD</p> <p>peut inclure des sous listes marquées elles-mêmes par <DL> et </DL></p>
<DT>	</DT>	<p>marqueur de début et de fin d'entrée dans une liste sans retrait</p> <p>un saut de ligne est effectué après</p>
<DD>	</DD>	<p>marqueur de début et de fin d'entrée dans une liste avec retrait</p> <p>un saut de ligne est effectué après</p>
		<p>défini la portion de texte qui doit avoir une couleur définie par x valant "#000000" en hexadécimal</p>
		<p>défini la portion de texte qui doit avoir une taille précisée par la valeur x, avec x variant de 1 (plus petit) à 7 (plus grand) par défaut la taille vaut 3</p>
<FORM>	</FORM>	<p>marqueur de début et de fin de formulaire</p> <p>deux paramètres principaux existent pour un formulaire</p> <p>METHOD="type" indiquant quel protocole HTTP il faut utiliser pour envoyer les données "Post" est le protocole à utiliser en général "get" étant l'alternative</p> <p>ACTION="adresse" indiquant l'endroit où les résultats de la saisie dans le formulaire doivent être envoyés, sous forme généralement d'une URL de @mail</p>

<HEAD>	</HEAD>	marqueur de début et de fin d'en-tête de page
<HTML>	</HTML>	marqueur de début et de fin de page, c'est à dire de début de fin de fichier
<HR>		écrit une ligne horizontale (Horizontal Rule) et provoque un saut de ligne avec un espacement avant et après Modifiable par SIZE= <i>nbpixels</i> Modifiable par ALIGN=" <i>right</i> " " <i>left</i> " " <i>center</i> " Modifiable par WIDTH=" <i>x%</i> " de l'écran
<Hx>	</Hx>	Modifie la taille du texte compris entre les deux tags selon une échelle variant de 1 (+grand) à 6 (plus petit) Insère de plus une ligne blanche après
		annonceur d'une entrée de liste qui sera dotée d'une puce ou d'un nombre (selon le type de liste ou) avec un retrait et un saut de ligne
<I>	</I>	met en italique peut s'insérer même autour d'un caractère dans un mot
<INPUT	TYPE=" <i>xx</i> ">	avec " <i>xx</i> " indiquant de quelle façon les données seront saisies par l'utilisateur <i>xx</i> valant " <i>text</i> ", " <i>password</i> ", " <i>radio</i> ", " <i>checkbox</i> ", " <i>submit</i> " et " <i>reset</i> "

	<p>marqueur simple d'insertion de l'image référencée par le paramètre SRC="xxx" avec xxx le nom du fichier image voulu N.B: la syntaxe ../dossier reste valable comme pour les liens</p> <p>Modifiable par ALIGN="top", "middle", "bottom" voire "right" et "left" Modifiable par VSPACE= <i>nbpixels</i> Modifiable par HSPACE= <i>nbpixels</i> Modifiable par WIDTH= <i>nbpixels</i> Modifiable par HEIGHT= <i>nbpixels</i> Modifiable par ALT= "texte en clair"</p>
 	<p>marqueur de début et de fin de liste numérotée; un saut de ligne avec ligne blanche est inséré avant et après encadre généralement des tag LI peut inclure des sous listes marquées elles-mêmes par ... ou ...</p>
<P>	<p>provoque un saut de ligne suivit de l'insertion d'une ligne vierge (fin de paragraphe)</p> <p>Le tag de fermeture </P> est optionnel</p>
<SELECT> </SELECT>	<p>encadrant l'ensemble des valeurs constituant une liste déroulante d'un formulaire, cChacune de ces valeurs étant précédée du tag <OPTION></p> <p>Si MULTIPLE est précisé une sélection multiple via la touche CTRL est autorisée</p>
<STRIKE> </STRIKE>	<p>défini la portion de texte qui sera barré</p>
	<p>défini la portion de texte en indice</p>
	<p>défini la portion de texte en exposant</p>

<TABLE >	</TABLE>	<p>marqueur de début et de fin de tableau Modifiable par BORDER=<i>nbpixels</i> Modifiable par WIDTH="<i>x%</i>" de l'écran Modifiable par CELSPACING=<i>nbpixels</i> Modifiable par CELLPADDING=<i>nbpixels</i></p>
<TD>	</TD>	<p>marqueur de début et de fin de cellule Le nombre de ces couples définira le nombre de colonnes du tableau Modifiable par WIDTH="<i>x%</i>" de l'écran Modifiable par ALIGN="<i>center</i>" ou "<i>right</i>" Modifiable par COLSPAN= <i>nbcellules</i> Modifiable par ROWSPAN= <i>nbcellules</i> Modifiable par NOWRAP Modifiable par VALIGN="<i>top</i>" ..."<i>baseline</i>"</p>
<TEXTAREA>	</TEXTAREA>	<p>Permet de définir une zone de texte dans un formulaire extensible "à volonté" par l'utilisateur lors de sa saisie Si ROWS et COLS sont précisés une taille en lignes et colonnes sera donnée</p>
<TH>	</TH>	<p>marqueur du titre du tableau qui est automatiquement centré et affiché en gras Modifiable par SIZE=<i>nbpixels</i> Modifiable par COLSPAN= <i>nbcellules</i> Modifiable par NOWRAP Modifiable par VALIGN="<i>top</i>" ..."<i>baseline</i>"</p>
<TITLE>	</TITLE>	<p>marqueur de début et de fin de titre de la page</p>
<TR>	</TR>	<p>marqueur de début et de fin de ligne de tableau</p>
		<p>marqueur de début et de fin de liste non numérotée; un saut de ligne avec ligne blanche est inséré avant et après encadre généralement des tag LI peut inclure des sous listes marquées elles-mêmes par ... ou ...</p>

LES TAGS HTML (PAR FONCTION)

Corps de fichier

<!--	-->	marqueur de début et de fin de commentaire
<BODY>	</BODY>	marqueur de début et de fin de corps d'une page modifiable par BGCOLOR="#000000" couleur de page BACKGROUND="xx" image de fond page LINK="#000000" coul. des liens ALINK="#000000" coul. de lien sélectionné VLINK="#000000" coul. de lien visité
<HEAD>	</HEAD>	marqueur de début et de fin d'en-tête de page
<HTML>	</HTML>	marqueur de début et de fin de page, c'est à dire de début de fin de fichier
<TITLE>	</TITLE>	marqueur de début et de fin de titre de la page

Mise en forme de texte

<code></code>	<code></code>	met en gras (Bold) peut s'insérer même autour d'un caractère dans un mot
<code>
</code>		provoque un saut de ligne Modifiable par CLEAR="all"
<code><BLINK></code>	<code></BLINK></code>	définit la portion de texte clignotante
<code>
</code>		provoque un saut de ligne
<code></code>	<code></code>	définit la portion de texte qui doit avoir une couleur définie par x valant "#000000" en hexadécimal
<code></code>	<code></code>	définit la portion de texte qui doit avoir une taille précisée par la valeur x, avec x variant de 1 (plus petit) à 7 (plus grand) par défaut la taille vaut 3
<code><Hx></code>	<code></Hx></code>	Modifie la taille du texte compris entre les deux tags selon une échelle variant de 1 (+grand) à 6 (plus petit) Insère de plus une ligne blanche après
<code><P></code>	<code></P></code>	provoque un saut de ligne suivi de l'insertion d'une ligne vierge (fin de paragraphe) Le tag de fermeture <code></P></code> est optionnel
<code><STRIKE></code>	<code></STRIKE></code>	définit la portion de texte qui sera barré
<code><SUB></code>	<code></SUB></code>	définit la portion de texte en indice
<code><SUP></code>	<code></SUP></code>	définit la portion de texte en exposant

Listes

<code><DL></code>	<code></DL></code>	marqueur de début et de fin de liste un saut de ligne avec ligne blanche est inséré avant et après encadre généralement des tag DT et DD peut inclure des sous listes marquées elles-mêmes par <code><DL></code> et <code></DL></code>
<code><DT></code>	<code></DT></code>	marqueur de début et de fin d'entrée dans une liste sans retrait un saut de ligne est effectué après
<code><DD></code>	<code></DD></code>	marqueur de début et de fin d'entrée dans une liste avec retrait un saut de ligne est effectué après
<code></code>		annonceur d'une entrée de liste qui sera dotée d'une puce ou d'un nombre (selon le type de liste <code></code> ou <code></code>) avec un retrait et un saut de ligne
<code><i></code>	<code></i></code>	met en italique peut s'insérer même autour d'un caractère dans un mot
<code></code>	<code></code>	marqueur de début et de fin de liste numérotée; un saut de ligne avec ligne blanche est inséré avant et après encadre généralement des tag LI peut inclure des sous listes marquées elles-mêmes par <code>...</code> ou <code>...</code>
<code></code>	<code></code>	marqueur de début et de fin de liste non numérotée; un saut de ligne avec ligne blanche est inséré avant et après encadre généralement des tag LI peut inclure des sous listes marquées elles-mêmes par <code>...</code> ou <code>...</code>

Lignes horizontales

<code><HR></code>		écrit une ligne horizontale (Horizontal Rule) et provoque un saut de ligne avec un espacement avant et après Modifiable par <code>SIZE=nbpixels</code> Modifiable par <code>ALIGN="right" "left" "center"</code> Modifiable par <code>WIDTH="x%"</code> de l'écran
-------------------------	--	---

Tableaux

<code><CAPTION></code>	<code></CAPTION></code>	marqueur de début et de fin de présentation de tableau, c'est à dire d'un titre hors tableau mais parfaitement centré sur lui. Doit être placé entre <code><Table></code> et la première rangée. Modifiable par <code>ALIGN="top" "bottom"</code>
------------------------------	-------------------------------	--

<code><TABLE ></code>	<code></TABLE></code>	marqueur de début et de fin de tableau Modifiable par <code>BORDER=nbpixels</code> Modifiable par <code>WIDTH="x%"</code> de l'écran Modifiable par <code>CELLSPACING=nbpixels</code> Modifiable par <code>CELLPADDING=nbpixels</code>
-----------------------------	-----------------------------	--

<code><TD></code>	<code></TD></code>	marqueur de début et de fin de cellule Le nombre de ces couples définira le nombre de colonnes du tableau Modifiable par <code>WIDTH="x%"</code> de l'écran Modifiable par <code>ALIGN="center" ou "right"</code> Modifiable par <code>COLSPAN= nbcellules</code> Modifiable par <code>ROWSPAN= nbcellules</code> Modifiable par <code>NOWRAP</code> Modifiable par <code>VALIGN="top" ..."baseline"</code>
-------------------------	--------------------------	--

<code><TH></code>	<code></TH></code>	marqueur du titre du tableau qui est automatiquement centré et affiché en gras Modifiable par <code>SIZE=nbpixels</code> Modifiable par <code>COLSPAN= nbcellules</code> Modifiable par <code>NOWRAP</code> Modifiable par <code>VALIGN="top" ..."baseline"</code>
-------------------------	--------------------------	--

<code><TR></code>	<code></TR></code>	marqueur de début et de fin de ligne de tableau
-------------------------	--------------------------	---

Liens

<code> xxx</code>	<code></code>	marqueur début / fin de lien
Ce tag nécessite que le paramètre "typelien" définisse quel lien il va falloir mettre en œuvre, en respectant les notations suivantes:		
<code>HREF="C:\SITE\MONFICH.HTM"</code>		un fichier HTM en local
<code>HREF="http://xxx"</code>		l'adresse xxx de sites sur le WEB
<code>HREF="ftp://xxx"</code>		adresse de dossiers ou fichiers sur un serveur ftp
<code>HREF="telnet://xxx"</code>		adressage de serveurs telnet
<code>HREF="gopher://xxx"</code>		adressage de dossiers ou de fichiers sur un serveur gopher
<code>HREF="newsxxx"</code>		adressage de Newsgroup
<code>HREF="mailto:xxx"</code>		Adressage d'adresse @Mail
Ce tag nécessite que la valeur de xxx définisse le texte qui renverra au lien lorsque l'on cliquera dessus		

<code></code>	<code></code>	marque de repère
Ce tag crée un repère "nom" référençable par une marque de type <code><A HREF.....></code> incluant la mention <code>#"nom"</code>		

Insertion d'Images

<code></code>	marqueur simple d'insertion de l'image référencée par le paramètre <code>SRC="xxx"</code> avec xxx le nom du fichier image voulu N.B: la syntaxe <code>../dossier</code> reste valable comme pour les liens
	Modifiable par <code>ALIGN="top", "middle", "bottom"</code> voire <code>"right"</code> et <code>"left"</code>
	Modifiable par <code>VSPACE= nbpixels</code>
	Modifiable par <code>HSPACE= nbpixels</code>
	Modifiable par <code>WIDTH= nbpixels</code>
	Modifiable par <code>HEIGHT= nbpixels</code>
	Modifiable par <code>ALT= "texte en clair"</code>

Formulaires

<code><FORM></code>	<code></FORM></code>	marqueur de début et de fin de formulaire
deux paramètres principaux existent pour un formulaire		
	<code>METHOD="type"</code>	indiquant quel protocole HTTP il faut utiliser pour envoyer les données "Post" est le protocole à utiliser en général "get" étant l'alternative
	<code>ACTION="adresse"</code>	indiquant l'endroit où les résultats de la saisie dans le formulaire doivent être envoyés, sous forme généralement d'une URL de @mail
<hr/>		
<code><INPUT</code>	<code>TYPE="xx"></code>	avec "xx" indiquant de quelle façon les données seront saisies par l'utilisateur xx valant "text", "password", "radio", "checkbox", "submit" et "reset"
<hr/>		
<code><SELECT></code>	<code></SELECT></code>	encadrant l'ensemble des valeurs constituant une liste déroulante d'un formulaire, cChacune de ces valeurs étant précédée du tag <code><OPTION></code> Si <code>MULTIPLE</code> est précisé une sélection multiple via la touche CTRL est autorisée
<hr/>		
<code><TEXTAREA></code>	<code></TEXTAREA></code>	Permet de définir une zone de texte dans un formulaire extensible "à volonté" par l'utilisateur lors de sa saisie Si <code>ROWS</code> et <code>COLS</code> sont précisés une taille en lignes et colonnes sera donnée

TAGS META (CHARSET).

Tags spéciaux pour les caractères nationaux

Syntaxe : <META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=ISO-8859-1">

Version : HTML 2.0

Où la placer ? : Entre les balises <HEAD> et </HEAD> (la première page de votre site).

Fonction : - Indique la langue (codage de vos documents) utilisée à l'intérieur de vos documents.

Langage	Charset
Africain	iso-8859-1, windows-1252
Albanais	iso-8859-1, windows-1252
Arabe	iso-8859-6
Basque	iso-8859-1, windows-1252
Bulgare	iso-8859-5
Biélorusse	iso-8859-5
Catalan	iso-8859-1, windows-1252
Croatie	iso-8859-2
Tchèque	iso-8859-2
Danois	iso-8859-1, windows-1252
Hollandais	iso-8859-1, windows-1252
Anglais	iso-8859-1, windows-1252
Espéranto *	iso-8859-3
Estonien *	iso-8859-10
Finlandais	iso-8859-1, windows-1252
Français	iso-8859-1, windows-1252

Galicien	iso-8859-1, windows-1252
Allemand	iso-8859-1, windows-1252
Grec	iso-8859-7
Hongrois	iso-8859-2
Islandais	iso-8859-1, windows-1252
Inuit *	iso-8859-10
Irlandais	iso-8859-1, windows-1252
Italien	iso-8859-1, windows-1252
Lapon *	iso-8859-10
Letton *	iso-8859-10
Lituanien *	iso-8859-10
Macédonien	iso-8859-5
Maltais *	iso-8859-3
Norvégien	iso-8859-1, windows-1252
Portugais	iso-8859-1, windows-1252
Roumain	iso-8859-2
Russe	koi-8-r, iso-8859-5
Ecossais	iso-8859-1, windows-1252
Serbe	iso-8859-5
Slovaque	iso-8859-2
Slovène	iso-8859-2
Espagnol	iso-8859-1, windows-1252
Suédois	iso-8859-1, windows-1252
Turc	iso-8859-9, windows-1254
Ukrainien	iso-8859-

TAGS META REFERENCEMENT.

Revisit After

Syntaxe : `<META NAME="Revisit-after" CONTENT="nombre">`

Nombre maxi de caractères : -

Compatibilité : Certains moteurs

Version : HTML 2.0

Où la placer ? : Entre les balises `<HEAD>` et `</HEAD>`

Fonctions :

- Permet de spécifier au spider (robot du moteur) de réindexer votre site suivant l'intervalle en jours donnée.

L'Erreur à ne pas commettre : Aucune, cette balise n'est pas indispensable.

Exemple : `<META NAME="Revisit-after" CONTENT="15 day">`

Title

Syntaxe : `<TITLE>Titre de votre site</TITLE>`

Nombre maxi de caractères : 100

Compatibilité : Tous moteurs

Version : HTML 2.0

Où la placer ? : Entre les balises `<HEAD>` et `</HEAD>`

Fonctions :

- Permet d'afficher le titre votre site dans la barre supérieure de votre navigateur.

- Lorsqu'un visiteur "bookmark" votre site c'est ce titre qui sera pris en compte comme intitulé de vos favoris ou signets.

- Intitulé pris en compte pas le moteur de recherche comme titre de votre site (première information vue par l'internaute recherchant un site sur un moteur ou annuaire).



Keywords

Syntaxe : <META NAME="Keywords" CONTENT="Vos mots clés ici">

Nombre maxi de caractères : 1000 Maxi

Compatibilité : Tous moteurs
Version : HTML 2.0

Où la placer ? : Entre les balises <HEAD> et </HEAD> (la première page de votre site)

Fonction :
- Indique aux moteurs de recherche les mots clés en rapport avec votre site.

L'Erreur à ne pas commettre : Ne répétez pas plusieurs fois de suite le même mot clé.

Note : Cette balise possède l'attribut LANG qui peut définir la langue utilisé pour votre site
(fr pour française, en pour Anglaise, en-us pour Américaine, de pour Allemande, it pour Italienne...)

Exemple : <META NAME="Keywords" LANG="fr" CONTENT="html, javascript, xml">

Description

Syntaxe : <META NAME="Description" CONTENT="Description de site...">

Nombre maxi de caractères : 200 Maxi

Compatibilité : Tous moteurs
Version : HTML 2.0

Où la placer ? : Entre les balises <HEAD> et </HEAD> (la première page de votre site)

Fonctions :
- Phrase prise en compte pas le moteur de recherche comme description de votre site.
- Les mots que constituent cette phrase sont aussi considérés comme des mots clés par la plupart des moteurs.

L'Erreur à ne pas commettre : Evitez une phrase qui n'a aucun sens.

Note : Cette balise possède l'attribut LANG qui peut définir la langue utilisé pour votre site.
(fr pour française, en pour Anglaise, en-us pour Américaine, de pour Allemande, it pour Italienne...).

Exemple : <META NAME="Description" CONTENT="bla bla...">

Robots

Syntaxe : `<META NAME="Robots" CONTENT="index, follow">`

Nombre maxi de caractères : -

Compatibilité : Certains moteurs
Version : HTML 2.0

Où la placer ? : Entre les balises <HEAD> et </HEAD>

Fonctions :

- Permet de spécifier au spider (robot du moteur) d'indexer (ou non) votre page et de suivre (ou non) les liens qui s'y trouvent

L'Erreur à ne pas commettre : Aucune, cette balise n'est pas indispensable.

Exemple : `<META NAME="Robots" CONTENT="index, follow">`

Exemple : `<META NAME="Robots" CONTENT="noindex, nofollow">`

Revisit After

Syntaxe : `<META http-equiv = "Content-Language" CONTENT="fr">`

Nombre maxi de caractères : -

Compatibilité : Certains moteurs
Version : HTML 2.0

Où la placer ? : Entre les balises <HEAD> et </HEAD>

Fonctions :

- Permet de spécifier au spider (robot du moteur) le langage de la page

L'Erreur à ne pas commettre : Aucune, cette balise n'est pas indispensable.

Exemple : `<META http-equiv = "Content-Language" CONTENT="fr, en ">`

NORMES HTML

Principe de tolérance

A partir du moment où on construit des pages HTML, ces pages vont être parcourues par des navigateurs divers et variés, tant au niveau des éditeurs que des versions. Or il est nécessaire que nos pages soient lues par le plus grand nombre correctement

HTML est un langage en perpétuel mouvement, et certains éditeurs n'ont pas hésité à créer de nouveaux tags compréhensibles uniquement par leur navigateur

Heureusement lorsqu'un navigateur rencontre un tag qu'il ne comprend pas, soit parce qu'il est mal écrit soit car il n'est pas standardisé, il l'ignore tout simplement sans générer de message d'erreur !

Tous les navigateurs du marché sont basés au minimum sur une version du langage HTML, et les sites Web s'entourent de précautions en affichant des messages du type "conçut pour tel navigateur version x.0"

HTML possède donc lui aussi des numéros de version permettant de définir précisément les tags utilisables ou non. C'est un consortium nommé W3C qui est chargé d'édicter les nouvelles normes du langage. Actuellement la dernière norme préconisée est la 3.2 le consortium venant à peine d'éditer la 4.0

On trouvera à l'adresse <http://www.w3.org> les spécifications actuelles du langage HTML ainsi que d'autres renseignements...

HTML ver 2.0 juin 1994:

Le [HTML 2.0](#) qui représentait la version courante du HTML en juin 1994, est le standard de base que tous les navigateurs d'aujourd'hui -- incluant les navigateurs-textes seulement -- devraient supporter. Le langage HTML 2.0 reflète la conception originale du HTML comme un langage indépendant des plates-formes et des logiciels pour afficher de l'information organisée (au lieu de spécifier exactement comment la page doit être affichée)



HTML ver 3.2 Mai 1997:

La [version HTML 3.0](#), sortie en 1995, a tenté de développer la version 2.0 en ajoutant des options comme les tableaux et un plus grand contrôle du texte autour des graphiques. En mai 1997, W3C fournissait la version [HTML 3.2](#), qui a été conçue pour refléter et standardiser les pratiques généralement acceptées. Ainsi, le HTML 3.2 inclut les étiquettes HTML 3.0 qui ont été adoptées par les développeurs de navigateurs comme Netscape et Microsoft.

Le langage HTML 3.2 offre les principales nouveautés suivantes :

- Introduire les tableaux
- Spécifier les frames

HTML ver 4.01 Décembre 99:

Le mois de juillet 97 donne naissance au langage 4.0 de ce même langage qui sera finalisé en décembre 99. Cette étonnante rapidité des normalisateurs du W3C est une nouveauté et elle a pour but de ne pas laisser les éditeurs de logiciels Netscape et Internet Explorer inventer leurs propres balises au gré des développements.

Le langage HTML 4.0 offre les principales nouveautés suivantes :

- Introduction des feuilles de style CSS pour traiter de l'alignement et du multi colonage.
- Permettre d'ajouter des étiquettes pour nommer les champs des formulaires
- S'ouvre aux langages de javascript – javascript – vbscript

HTML ou XHTML

On peut voir souvent dans la littérature que à côté de HTML, d'autres sigles paraissent, et semblent de plus en plus utilisés de nos jours.

Le eXtensible HyperText Markup Language (XHTML) a été officialisé en 2000.

Il s'agit d'une reformulation du HTML sous les règles du eXtensible Markup Language (XML). Le XHTML demande un effort supplémentaire de rigueur, mais pour l'essentiel ressemble de très près au HTML. Un des très grands avantages de choisir d'utiliser le XHTML: puisque il est fondé sur le XML, tous les outils de traitement de XML deviennent d'un seul coup disponibles, permettant une variété impressionnante de manipulations permettant de reformater l'information concentrée dans le document.



Le squelette d'une page XHTML - HTML

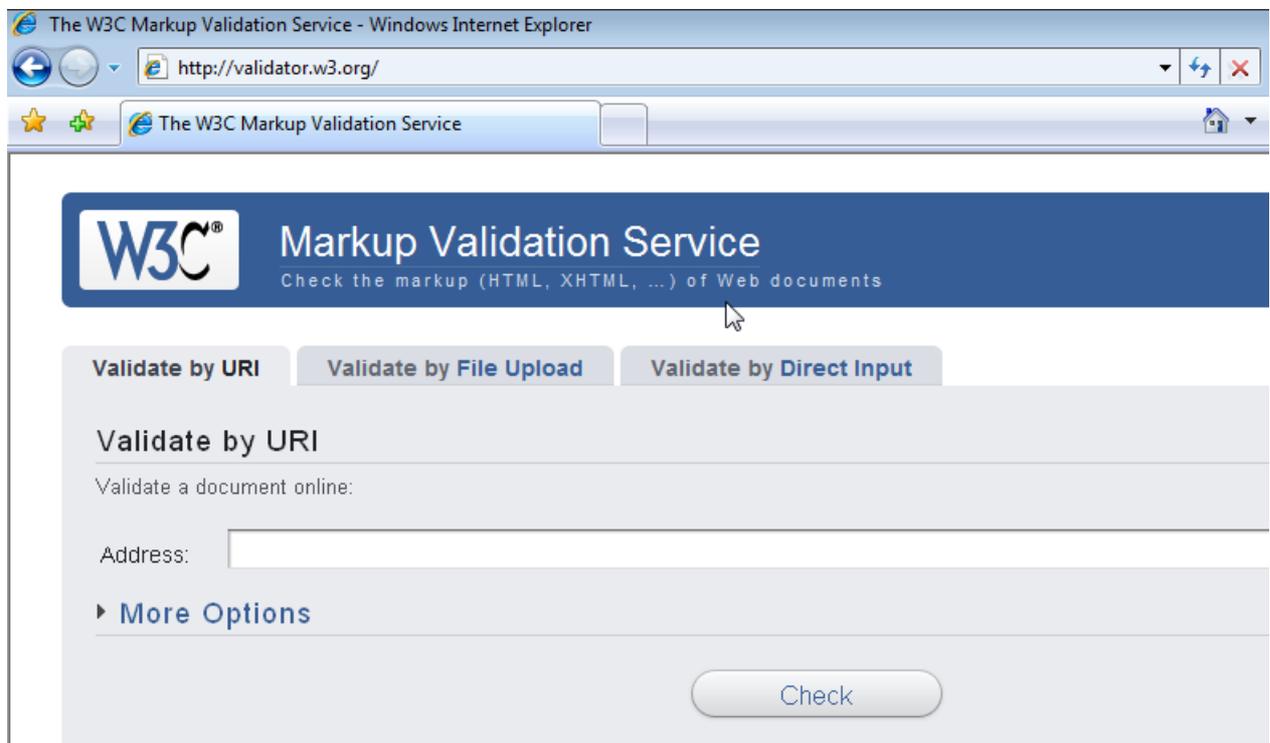
Puisqu'un document XHTML ne constitue qu'un cas particulier de document XML, on doit révéler au début du document quelles balises seront utilisées.

La déclaration `<!DOCTYPE>` sert justement à indiquer cela.

- Le **Document Type Definition** (DTD), qui précise au navigateur quelle version de XHTML vous utilisez :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Vérification W3c:



The W3C Markup Validation Service - Windows Internet Explorer

http://validator.w3.org/

The W3C Markup Validation Service

W3C Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI | Validate by File Upload | Validate by Direct Input

Validate by URI

Validate a document online:

Address:

▶ More Options

Check

MULTIMEDIA ET HTML

Lien simple

La seule solution reconnue universellement c'est la balise <HREF> mais elle ne permet aucun contrôle sur la manière dont le fichier s'exécutera sur le navigateur (elle est très minimaliste). La standardisation n'existe pas théoriquement dans ce domaine est elle n'est réalisée que de fait par le développement de telle ou telle norme de fait...

Un tag passe partout sera donc

marqueur simple d'insertion de son ou de vidéo référencé par le paramètre HREF="xxx" avec xxx le nom du fichier son ou vidéo désiré

Plug-in

La lecture d'un document multimédia par le navigateur n'est pas si simple. Il doit disposer d'un plug-in (greffon en français) adaptés comme :

Macromedia Flash,
Real Player,
QuickTime,
Windows Media Player...

À défaut de plug-in adéquat, le visiteur ne verra rien apparaître si ce n'est un texte alternatif s'il a été prévu et peut-être un lien vers le téléchargement du plug-in adapté.

Plug-in Flash

Flash appartient à Adobe (Macromedia). Il reconnaît les formats **.swf**

Plug-in QuickTime

[QuickTime](#) appartient à Apple et fonctionne aussi bien sur Mac que sur PC. Il reconnaît plusieurs formats **.qtl, .mov, .mpg, .mp4 et .mp3**.

Plug-in Windows Media Player

Windows Media Player reconnaît plusieurs formats: **.wmv, .wma, .avi, .mpg, .mp3 et .mp4**. Il est disponible nativement sur les systèmes Windows.



Les Balises possibles

Deux Balises HTML spécifiques au son ou à la vidéo (multimédia...) existent:

EMBED.

OBJECT

La balise <EMBED> est un élément obsolète et ne passe pas la validation XHTML.

La balise <OBJECT> est tout à fait adaptée et valide mais donne parfois des mauvais résultats d'affichage surtout dans certains navigateurs.

	Netscape 2	Netscape 3, 4.x	FireFox 2.x, 3x	I.Explorer 3, 4.x, 5.x	I.Explorer 6.x, 7.x
<HREF>	Oui	Oui	Oui	Oui	Oui
<BGSOUND>	Non	Non	Non	Oui	Oui
	Non	Non	Non	Oui	Oui
<EMBED>	Non	Oui	Oui	Oui	Oui
<NO EMBED>	Non	Oui	Oui	Oui	Oui
<OBJECT>	Non	Oui (v4)	Oui	Oui (v4 et v5)	Oui

Balises "propriétaires":

<BGSOUND> Microsoft et Microsoft

<EMBED> Netscape, puis reconnue par Internet Explorer depuis sa version 3x

Si l'on souhaite passer la norme XHTML, et pouvoir valider ses pages auprès du w3C, il faudrait utiliser la balise OBJECT, seule balise préconisée en html 4.0 par le W3C. Cependant IE ne fonctionne pas toujours correctement avec la balise object, et marche parfois mieux avec l'ancienne balise EMBED...

Différentes écritures peuvent donc subsister pour assurer la compatibilité entre navigateurs... notamment, à la date d'aujourd'hui entre IE 6-7 et Firefox 2-3

Les paramètres sont passés à la balise EMBED par des paires de valeur

nom="valeur"

Des paramètres peuvent être passés à la balise OBJECT via une balise PARAM .

<PARAM name="" value="xxxx"/>

N.B: Mais ces balises PARAM ne sont pas normalisées, car ils dépendent des plug-in utilisés par le navigateur qui interprète la page HTML, hors à priori on ne peut pas connaître le plug-in installé sur le client !

Balise Embed

Utilisation	Inclut, dans une page Web, tout type d'objet (texte, image, audio, vidéo...)
Balises de début/de fin	Nécessaire/facultatif
Attributs	<p>src="..." (obligatoire) l'URL source du fichier multimédia à afficher</p> <p>type="..." (facultatif) type MIME définissant le format des données à afficher</p> <p>pluginspage = "..." (facultatif) l'URL permettant de télécharger le plugin en cas de besoin</p> <p>et pour la mise en forme</p> <p>alt="..." : texte de remplacement à afficher</p> <p>height="..." hauteur d'affichage</p> <p>width="..." largeur d'affichage</p> <p>border="..." affiche une bordure autour de l'objet</p>
Attributs IE	<p>Paramètre optionnel. Ces paramètres définissent des états alternatifs en prenant comme valeur TRUE ou FALSE.</p> <p>AUTOSTART : Spécifie si le fichier source est joué automatiquement (TRUE) ou non (FALSE) (défaut : TRUE).</p> <p>AUTOLOAD : Spécifie si le chargement de la vidéo s'effectue automatiquement (TRUE) ou non (FALSE) (défaut : TRUE).</p> <p>LOOP : Spécifie le nombre de fois où la vidéo est jouée (si la valeur est -1 le fichier sera joué indéfiniment).</p> <p>SHOWCONTROLS : Indique si le panneau de contrôle s'affiche (TRUE) ou non (FALSE) (défaut : TRUE).</p>

exemple: **multimedia-00.html**:

```
<embed src="sons-videos/fond-sonore.mp3" width="300" height="40"
autostart="true" loop="false">
```

```
<noembed> probleme de lecture de son
```

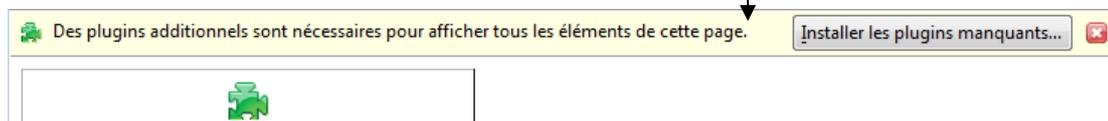
```
</noembed>
```

```
</embed>
```

peut donner si les plug-ins sont installés



ou pour Firefox dans le cas contraire...



Firefox "par défaut" utilise le plug-in de Apple QuickTime pour les fichiers mp3, si celui-ci n'est pas chargé, alors il est demandé...

Il ne faut pas en déduire autre chose que cela... si on veut utiliser un plug-in spécifique, il faut alors ajouter l'instruction Type Mime, avec une valeur particulière...

chez les développeurs de mozilla, on trouve une liste de MIME <http://plugindoc.mozdev.org/winmime.php> dans laquelle on voit que

si on donne un nom de type de fichier

audio/mp3	MPEG Layer 3 Audio	QuickTime Crescendo Player
-----------	--------------------	---

ce qui fait que sous firefox on utilisera le plug-in Quicktime, et sous IE on utilisera le plug-in Windows Média Player

alors que

application/x-mplayer2	Windows Media	Windows Media Player
------------------------	---------------	--------------------------------------

ou encore plus précis

application/x-ms-wmp	Windows Media (Force v11 Player)	Windows Media Player 11
----------------------	----------------------------------	---

ce qui fait que sous firefox et IE on utilisera le plug-in Windows Média Player

Si donc on rajoute **TYPE="application/x-ms-wmp"** on jouera le morceau MP3 sous IE et Firefox avec Windows Media Player 11... (mais cela ne marchera pas avec une machine Linux ou Mac...)

exemple: **multimedia-01.html**:

```
<embed src="sons-videos/fond-sonore.mp3" TYPE="application/x-ms-wmp"
width="300" height="40" autostart="true" loop="false">
  <noembed> probleme de lecture de son
</noembed>
</embed>
```

Ainsi, pour jouer un "fond" sonore sur une page, on utilisera alors la balise avec les paramètres suivants

exemple: **multimedia-02.html**:

```
<embed src="sons-videos/fond-sonore.mp3" TYPE="application/x-ms-wmp"
autostart="true" loop="true" showControls="false">
  <noembed> probleme de lecture de son
</noembed>
</embed>
```

N.B: noter qu'il ne faut plus indiquer la taille en pixel de l'objet, et que beaucoup de paramètres dépendants du plug-in existent.... Ainsi Firefox ne cache pas le lecteur... Les écritures peuvent varier et être incompatibles entre navigateurs-plug-ins différents...

N.B: pour toutes ces raisons, **EMBED** est une balise à ne plus utiliser, sauf à vouloir être compatible avec de très anciens navigateurs...

Balise Object

Utilisation	Inclut, dans une page Web, tout type d'objet (texte, image, audio, vidéo...)
Balises de début/de fin	Nécessaire/nécessaire
Attributs	<code>data="..."</code> (obligatoire) l'URL source du fichier multimédia à afficher <code>type="..."</code> (obligatoire) type MIME définissant le format des données à afficher <code>classid="..."</code> référence l'implémentation du contrôle ACTIVEX souhaité par son CLSID. Permet de cibler le player à utiliser en environnement MICROSOFT-WINDOWS-IE <code>codebase="..."</code> l'URL permettant de télécharger le plugin en cas de besoin <code>codetype="..."</code> type de contenu Internet du code, à utiliser à la place de type lorsque l'option classid n'est pas utilisée. et pour la mise en forme <code>standby="..."</code> message affiché au cours du chargement <code>height="..."</code> hauteur de l'objet <code>width="..."</code> largeur de l'objet <code>border="..."</code> affiche une bordure autour de l'objet

Exemple avec data : **multimedia-03.html**:

```
<object data=" sons-videos/fond-sonore.mp3" type="audio/mpeg" width="300" height="40">
</object>
```

Firefox déclenche Quicktime (si ce plug-in est présent)

Mais IE ne marche pas avec ce code, il faut lui ajouter au minimum la ligne

```
<param name="src" value=" sons-videos/fond-sonore.mp3" />
```

Comme dans **multimedia-03b.html**

```
<object data=" sons-videos/fond-sonore.mp3" type=" audio/mpeg " width="300" height="40">
<param name="src" value=" sons-videos/fond-sonore.mp3" />
</object>
```

Et dans ce cas autant ajouter d'autres paramètres intéressants

```
<param name="autoplay" value="false" />
```

```
<param name="loop" value="false" />
```

Comme dans **multimedia-03c.html**

```
<object data=" sons-videos/fond-sonore.mp3" type="audio/mpeg" width="300" height="40">
```

```
<param name="src" value=" sons-videos/fond-sonore.mp3" />
```

```
<param name="autoplay" value="false" />
```

```
<param name="loop" value="false" />
```

```
</object>
```

cette écriture est conforme aux normes



Jump To: [Congratulations](#) · [Icons](#)

This document was successfully checked as XHTML 1.0 Transitional!

N.B: les balises `<param>` s'ont interprétées différemment suivant le lecteur (quicktime, real, mplayer etc...), il y a même un risque de "plantage" si le lecteur ne gère pas le `<param>`.

NB: Ici, c'est quicktime qui est appelé par défaut sur Firefox et Netscape, Windows Media Player Sur IE... Si on veut forcer Windows Media Player, on peut alors changer le type MIME par **type=" application/x-mplayer2"** Comme dans **multimedia-03d.html**

On peut noter que IE préconise l'écriture suivante avec classid :

Comme dans **multimedia-04.html**

```
<object type=" application/x-oleobject " classid="clsid:6BF52A52-394A-11D3-B153-00C04F79FAA6" width="300" height="40">
```

```
<param name="url" value="sons-videos/fond-sonore.mp3" />
```

```
<param name="autostart" value="true" />
```

```
<param name="loop" value="false" />
```

```
</object>
```

par contre cela ne marche pas avec Firefox ...

type MIME :

Le type MIME (Multipurpose Internet Mail Extensions) est utilisé d'une part pour typer les documents attachés à un courrier mais aussi pour typer les documents transférés par le protocole HTTP. Ainsi lors d'une transaction entre un serveur web et un navigateur internet, le serveur web envoie en premier lieu le type MIME du fichier envoyé au navigateur, afin que ce dernier puisse savoir de quelle manière afficher le document.

Un type MIME est constitué de la manière suivante :

Content-type: type_mime_principal/sous_type_mime

Une image GIF a par exemple le type MIME suivant :

Content-type: image/gif

N.B: il est important de comprendre que le type MIME, normalement, n'indique pas le plug-in à utiliser, mais donne une indication sur la nature du fichier téléchargé...

Le navigateur est censé lancer un plug-in selon ses préférences, et même interpréter le fichier différemment de l'indication fournie par le type-mime en cas de problèmes...

Il peut y avoir des exceptions parfois lorsque le type indique une application...

Type MIME	Type de fichier	Extension associée
application/atom+xml	Fichiers au format ATOM	atom
application/iges	Fichiers CAS	iges
application/javascript	Fichiers Javascript	js
application/dxf	Fichiers AutoCAD	dxf
application/mp4	Fichiers MPEG4	mp4
application/octet-stream	Fichiers binaires non interprétés	bin
application/msword	Fichiers bureautique au format Microsoft Word	doc
application/pdf	Fichiers Adobe Acrobat	pdf
application/postscript	Fichiers PostScript	ai,eps,ps
application/rtf	Format de texte enrichi	rtf
application/sgml	Fichiers SGML	sgml
application/x-shockwave-flash	Fichiers Flash	swf
application/vnd.ms-excel	Fichiers tableur au format Microsoft Excel	xls
application/vnd.ms-powerpoint	Fichiers diaporama au format Microsoft Powerpoint	ppt
application/xml	fichier XML	xml
application/x-tar	Fichiers compressés tar	tar
application/zip	Fichiers compressés ZIP	man
audio/basic	Fichiers audio basiques	au,snd

audio/mpeg	Fichiers audio MPEG	mpg,mp3
audio/mp4	Fichiers audio MPEG-4	mp4
audio/x-aiff	Fichiers audio AIFF	aif,aiff,aifc
audio/x-wav	Fichiers audio Wave	wav
image/gif	Images gif	man
image/jpeg	Images JPEG	jpg,jpeg,jpe
image/png	Images PNG	png
image/x-portable-bitmap	Fichiers Bitmap PBM	pbm
image/x-portable-graymap	Fichiers Graymap PBM	pgm
image/x-portable-pixmap	Fichiers Pixmap PBM	ppm
multipart/x-zip	Fichiers archive zip	zip
multipart/x-gzip	Fichiers archive GNU zip	gz,gzip
text/css	Feuille de style	css
text/csv	Fichiers texte avec séparation des valeurs	csv
text/html	Fichiers HTML	htm,html
text/plain	Fichiers texte sans mise en forme	txt,g,h,c,cc,hh,m,f90
text/richtext	Fichiers texte enrichis	rtx
text/rtf	Fichiers texte au format Rich Text Format	rtf
text/tab-separated-value	Fichiers texte avec séparation des valeurs	tsv
text/xml	Fichiers XML	xml
video/h264	Vidéos H.264	h264
video/dv	Vidéos au format DV	dv
video/mpeg	Vidéos MPEG	mpeg,mpg,mpe
video/quicktime	Vidéos QuickTime	qt,mov
video/avi	Vidéos Microsoft Windows	avi
video/msvideo	Vidéos Microsoft Windows	avi

une liste des MIME existe en <http://www.iana.org/assignments/media-types/>

on peut aussi essayer de savoir quel plug-in est associée à quel type-mime ...

Extensions	Type MIME	Plug-in possibles (Firefox-IE)
.avi	video/avi	QuickTime - Windows Media Player
.css	text/css	Géré nativement
.gif	image/gif	Géré nativement
.htm / .html	text/html	Géré nativement
.ico / .cur	image/x-icon	Géré nativement
.jpg / .jpeg	image/jpeg	Géré nativement
.js	text/javascript	Géré nativement
.mid / .midi	audio/midi	QuickTime - Windows Media Player

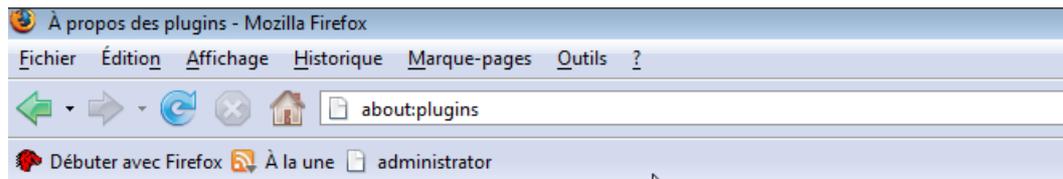


.mov	video/quicktime	QuickTime - Windows Media Player
.mp3	audio/mp3	QuickTime - Windows Media Player
.mp3	audio/mpeg	QuickTime - Windows Media Player
.mp3	application/x-mplayer2	Windows Media Player
.mpg / .mpeg	video/mpeg	VLC Media Player - Windows Media Player
.ogg	application/x-ogg	VLC Media Player
.pdf	application/pdf	Acrobat Reader
.png	image/png	Géré nativement
.ra, .ram	audio/vnd.rn-realaudio	RealAudio
.rm	application/vnd.rn-realmedia	RealMedia
.swf	application/x-shockwave-flash	Macromedia Flash Player
.tif / .tiff	image/tif	QuickTime
.txt	text/plain	Géré nativement
.wav	audio/wav	QuickTime - Windows Media Player
.xml	application/xhtml+xml	Géré nativement

on peut aussi essayer de savoir quel plug-in est installé sur un navigateur ...

sous Firefox on tape comme URL

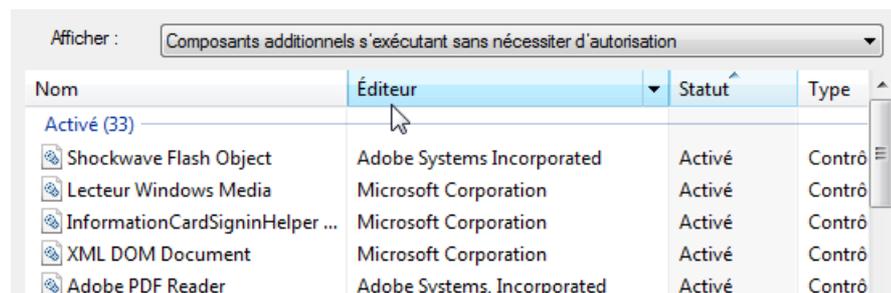
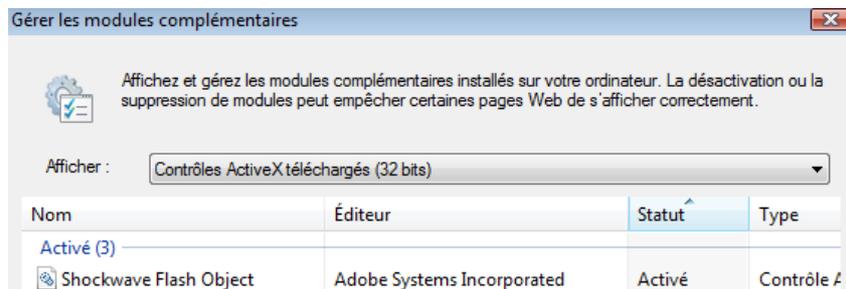
about:plugins



Plugins installés

Vous trouverez plus d'informations sur les plugins de votre navigateur sur mozilla.org.
De l'aide pour l'installation des plugins est disponible sur plugindoc.mozdev.org.

sous IE cela dépend des versions... sous IE7, Outils / Options Internet / Programmes / Gérer les modules complémentaires



classid et codebase :

Si on utilise dans le code HTML les attributs classid et codebase pour l'élément <object>, il faut savoir que chaque éditeur fournit le code 'object' complet pour sa technologie, avec le bon code activeX qui correspond à la bonne version du player.

N.B: Il est à noter que pour Windows Media Player, la commande codebase est déconseillée à partir de la version 7.0..., et que pour realplayer, elle est inexistante...

Plug-in Flash

classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"

codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=6,0,40,0"

Windows Media Player 7-9-10-11

classid="clsid:6BF52A52-394A-11D3-B153-00C04F79FAA6"

version 6.4 clsid:22D6F312-B0F6-11D0-94AB-0080C74C7E95
version 6.0 clsid:05589FA1-C356-11CE-BF01-00AA0055595A

codebase=http://activex.microsoft.com/activex/controls/mplayer/en/nsmp2inf.cab#Version=6,4,5,715

QuickTime

classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B"

codebase=http://www.apple.com/qtactivex/qtplugin.cab

Real Player

classid="clsid:CFCDA03-8BE4-11cf-B84B-0020AFBCCFA"

Balise Object et Embed

en résumé il faut choisir...

- ou on veut écrire selon la règle, et on utilise uniquement Object
- ou on veut être compatible avec de vieux navigateurs, et on utilise Embed
- ou on veut laisser le type de plug-in libre, et cela sera les associations Windows-Media Player pour IE, et QuickTime pour Firefox Netscape
- ou on cible un plug-in par le type MIME... genre
- ou on veut écrire en utilisant toutes les possibilités de Windows Media Player, en suivant la norme Windows, et alors il faut indiquer avec Object le paramètre CLSID, mais pour que Firefox fonctionne il faut alors imbriquer les balises Object et Embed ...

Comme dans **multimedia-05.html**

```
<object type="application/x-oleobject" classid="clsid:6BF52A52-394A-11D3-B153-00C04F79FAA6" width="300" height="40">
    <param name="url" value="sons-videos/fond-sonore.mp3" />
    <param name="autostart" value="true" />
    <param name="loop" value="false" />
    <embed src="sons-videos/fond-sonore.mp3" type="application/x-mplayer2"
        autostart="false" loop="false">
        <noembed> probleme de lecture de son
        </noembed>
    </embed>
</object>
```

Video AVI

L'intégration d'un fichier AVI peut s'effectuer de différentes façons.

Comme dans **multimedia-06.html**

```
<object type="video/avi" data="sons-videos/oiseau.avi" width="320"
height="255">
    <param name="src" value="sons-videos/oiseau.avi">
    <param name="autostart" value="false" />
    <param name="loop" value="false" />
</object>
```

NB: Ici, c'est quicktime qui est appelé par défaut sur Firefox et Netscape, Windows Media Player Sur IE... Si on veut forcer Windows Media Player, on peut alors changer le type MIME par **type=" application/x-mplayer2"**

Comme dans **multimedia-06b.html**

et les mêmes montages imbriqués peuvent être réalisés....

```
<OBJECT width="300" height="270" CLASSID="CLSID:6BF52A52-394A-11d3-B153-
00C04F79FAA6" type="application/x-oleobject">
    <PARAM NAME="URL" VALUE=" sons-videos/oiseau.avi ">
    <PARAM NAME="SendPlayStateChangeEvents" VALUE="True">
    <PARAM NAME="AutoStart" VALUE="true">
    <PARAM name="uiMode" value="full">
    <PARAM name="enabled" value="true">
    <PARAM name="PlayCount" value="9999">
    <EMBED type="application/x-mplayer2" pluginspage =
"http://www.microsoft.com/Windows/MediaPlayer/"
SRC=" sons-videos/oiseau.avi " width="300" height="270" AutoStart="true"
autosize="0" transparentatStart="true" animationatStart="true"
showControls="true">
    </EMBED>
</OBJECT>
```

Incorporation de flash

L'intégration d'un fichier flash peut s'effectuer simplement, en effet le lecteur est ... unique, chez ADOBE



Comme dans **multimedia-07.html**

```
<object type="application/x-shockwave-flash" data="sons-videos/demo-  
flash.swf" width="400" height="400">  
    <param name="URL" value="sons-videos/demo-flash.swf" />  
    <param name="movie" value="sons-videos/demo-flash.swf" />  
    <param name="wmode" value="transparent" />  
    <param name="quality" value="high" />  
</object>
```

Paramètres flash player

L'utilisation de flash player permet certains paramètres, ...

http://kb.adobe.com/selfservice/viewContent.do?externalId=tn_12701

Cependant raisonnablement on peut trouver les valeurs suivantes :

- width - Specifies the width of the movie in either pixels or percentage of browser window.
- height - Specifies the height of the movie in either pixels or percentage of browser window.
- classid - Identifies the ActiveX control for the browser.

- `codebase` - Identifies the location of the Flash Player ActiveX control so that the browser can automatically download it if it is not already installed. (See example code in TechNote tn_4150 for the correct value.)
- `movie` (`param`) - Specifies the location (URL) of the movie to be loaded.
- `play` - Possible values: `true`, `false`. Specifies whether the movie begins playing immediately on loading in the browser. The default value is `true` if this attribute is omitted.
- `loop` - Possible values: `true`, `false`. Specifies whether the movie repeats indefinitely or stops when it reaches the last frame. The default value is `true` if this attribute is omitted.
- `menu` - Possible values: `true`, `false`.
 - `true` displays the full menu, allowing the user a variety of options to enhance or control playback.
 - `false` displays a menu that contains only the Settings option and the About Flash option.
- `quality` - Possible values: `low`, `high`, `autolow`, `autohigh`, `best`.
 - `low` favors playback speed over appearance and never uses anti-aliasing.
 - `autolow` emphasizes speed at first but improves appearance whenever possible. Playback begins with anti-aliasing turned off. If the Flash Player detects that the processor can handle it, anti-aliasing is turned on.
 - `autohigh` emphasizes playback speed and appearance equally at first but sacrifices appearance for playback speed if necessary. Playback begins with anti-aliasing turned on. If the actual frame rate drops below the specified frame rate, anti-aliasing is turned off to improve playback speed. Use this setting to emulate the View > Antialias setting in Flash.
 - `medium` applies some anti-aliasing and does not smooth bitmaps. It produces a better quality than the Low setting, but lower quality than the High setting.
 - `high` favors appearance over playback speed and always applies anti-aliasing. If the movie does not contain animation, bitmaps are smoothed; if the movie has animation, bitmaps are not smoothed.
 - `best` provides the best display quality and does not consider playback speed. All output is anti-aliased and all bitmaps are smoothed.
- `scale` - Possible values: `showall`, `noborder`, `exactfit`.
 - `default` (Show all) makes the entire movie visible in the specified area without distortion, while maintaining the original aspect ratio of the movie. Borders may appear on two sides of the movie.
 - `noorder` scales the movie to fill the specified area, without distortion but possibly with some cropping, while maintaining the original aspect ratio of the movie.

- `exactfit` makes the entire movie visible in the specified area without trying to preserve the original aspect ratio. Distortion may occur.
- `align` (attribute for Object) - Possible values: l, t, r, b.
 - Default centers the movie in the browser window and crops edges if the browser window is smaller than the movie.
 - l (left), r (right), t (top), and b (bottom) align the movie along the corresponding edge of the browser window and crop the remaining three sides as needed.
- `salign` - Possible values: l, t, r, b, tl, tr, bl, br.
 - l, r, t, and b align the movie along the left, right, top or bottom edge, respectively, of the browser window and crop the remaining three sides as needed.
 - tl and tr align the movie to the top left and top right corner, respectively, of the browser window and crop the bottom and remaining right or left side as needed.
 - bl and br align the movie to the bottom left and bottom right corner, respectively, of the browser window and crop the top and remaining right or left side as needed.
- `wmode` - Possible values: window, opaque, transparent. Sets the Window Mode property of the Flash movie for transparency, layering, and positioning in the browser.
 - window - movie plays in its own rectangular window on a web page.
 - opaque - the movie hides everything on the page behind it.
 - transparent - the background of the HTML page shows through all transparent portions of the movie, this may slow animation performance.
- `bgcolor` - [*hexadecimal RGB value*] in the format #RRGGBB . Specifies the background color of the movie. Use this attribute to override the background color setting specified in the Flash file. This attribute does not affect the background color of the HTML page.
- `base` - . or [*base directory*] or [*URL*]. Specifies the base directory or URL used to resolve all relative path statements in the Flash Player movie. This attribute is helpful when your Flash Player movies are kept in a different directory from your other files.
- `flashvars` - Possible values: *variable to pass to Flash Player*. Requires Macromedia Flash Player 6 or later.
 - Used to send root level variables to the movie. The format of the string is a set of name=value combinations separated by '&'.

Vidéo swf dans...flash

L'intégration d'un fichier flash "englobant" une video, peut s'effectuer simplement, en effet la vidéo est traitée comme un fichier flash quelconque...

Comme dans **multimedia-08.html**

```
<object          type="application/x-shockwave-flash"          data="sons-
videos/cavalcade.swf" width="400" height="400">
  <param name="movie" value="sons-videos/cavalcade.swf" />
  <param name="wmode" value="transparent" />
  <param name="quality" value="high" />
</object>
```

Vidéo .flv et player flash flv

Si vous encodez votre fichier en .flv, pour bénéficier du téléchargement progressif, pour visualiser la vidéo, il ne faudra plus la traiter comme un fichier flash classique, mais utiliser un player flv...

Ce player (ex : FLVPlayer ou player_flv_maxi) doit être installé dans un dossier de votre site web.

on le récupère par exemple sur



et le player est donc un fichier

 player_flv_maxi.swf	08/04/2009 18:34	Shockwave Flash Object	13 Ko
---	------------------	------------------------	-------

La page web devra quand à elle contenir le code suivant :

Comme dans **multimedia-09.html**

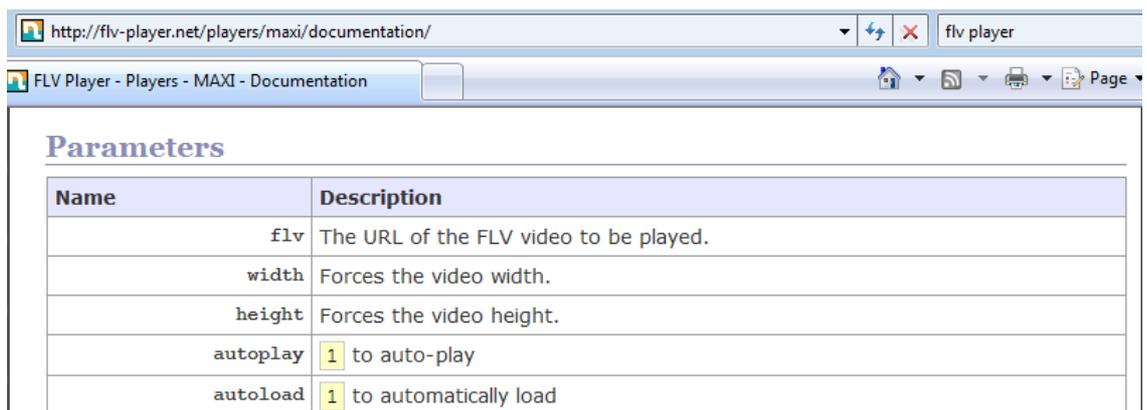
```
object type="application/x-shockwave-flash" data="player_flv_maxi.swf" width="400"
height="400">
  <param name="movie" value="player_flv_maxi.swf" />
  <param name="FlashVars" value="flv=sons-videos/oiseau.flv" />
</object>
```

on affiche le player (au format flash) via le plug-in flash classique...

et on passe les paramètres voulus à ce player via la variable FlashVars.. (au minimum, la valeur **flv=.....** permettant de donner le source de la video en flv=

on peut affiner les réglages du player

sur le site du player, une documentation existe...



Name	Description
flv	The URL of the FLV video to be played.
width	Forces the video width.
height	Forces the video height.
autoplay	1 to auto-play
autoload	1 to automatically load

...

showstop	1 to show the STOP button.
showvolume	1 to show the VOLUME button.
showtime	1 to show the TIME button, 2 to show the remaining time by default
showplayer	Player bar display mode : autohide , always or never .
showloading	Loading bar display mode : autohide , always or never .
showfullscreen	1 to show fullscreen button (requires Flash Player 9.0.16.60 or newer)
showswitchsubtitles	1 to show the button showing/hiding subtitles
showmouse	Display of the mouse cursor : always , autohide , never .
loop	1 to loop.
startimage	The URL of the JPEG file (not progressive) to be shown before loading the video

Comme dans **multimedia-09b.html**

```
value="flv=sons-videos/cavalcade.flv&amp;autoplay=1&amp;showstop=1&amp;showtime=1
&amp;showvolume=1" />
```

FEUILLES DE STYLE - CSS

Historique des CSS :

HTML conçu au départ comme langage universel pour décrire le contenu et la structure d'un document, et non pas leur aspect, s'est vu progressivement rajouter bon nombre de fonctionnalités, notamment par la balise

Mais au lieu d'étendre sans cesse les possibilités du langage HTML, il a été pensé plus sage de décrire la présentation d'un document dans un fichier à part.

Ainsi HTML retrouve sa fonctionnalité première, à savoir décrire la structure et le contenu d'un document, indépendamment de sa mise en forme, qui se retrouve stockée dans un autre document

le **W3C** est pratiquement à l'initiative de cette solution, même si, comme d'habitude, il n'y a pas unanimité sur la méthode pour décrire la présentation d'un document HTML dans une feuille de style, d'autant plus que ces feuilles étant désormais des documents à part, grande est la tentation d'inventer des normes un peu "propriétaires"...

La définition officielle est disponible auprès du W3C à l'adresse <http://www.w3w.org>

CSS1	Cascading Style Sheets	version 1	Décembre 1996
CSS2	Cascading Style Sheets	version 2	Mai 1998
CSS3	Cascading Style Sheets	version 3	En cours ...

Les feuilles de style sont supportées par les deux grands navigateurs depuis leurs versions 3.x respectives mais de façons incomplètes et/ou différentes ...

**N.B: AUJOURD'HUI ENCORE AUCUN NAVIGATEUR NE RESPECTE TOTALEMENT CSS1 , ET ENCORE MOINS CSS2 !
A MEDITER LORSQUE ON VERRA LES DIVERSES BALISES**

Une feuille de style c'est donc un modèle de mise en forme applicable à un texte, comme par exemple :

- Gras, italique, souligné
- espacement entre les lignes
- largeur de la marge droite



Avantages des feuilles de style :

Puisque une feuille de style sépare les caractéristiques de présentation du contenu, elle peut permettre de regrouper une série de mise en forme commune et donc faciliter l'homogénéisation de présentation d'un site

Une modification de la feuille de style permettra une modification de l'affichage de toutes les pages HTML qui sont fondées dessus !

Puisque la mise en forme est regroupée dans un fichier distinct, il ne devraient plus y avoir de nouvelles balises HTML spécifiques à telle ou telle mise en forme sur tel ou tel navigateur (du moins en ce qui concerne le HTML "pur", c'est à dire le langage structurel de la page)

Pourquoi "En Cascade" :

Les feuilles de styles sont dites en cascades (**CSS** pour **Cascading Style Sheet**) car il existe une notion de priorité dans les mises en formes, de manière à ce que les navigateurs puissent décider en cas de problème, quel style utiliser

De plus, il est possible d'intégrer les feuilles de style de plusieurs manières dans une page HTML, essentiellement de quatre manières :

- Stocker les styles dans un fichier à part via une balise <LINK>
on parle de style associés ou "linking"
- Stocker les styles dans l'entête de la page HTML entre des balises <LINK> et </LINK>
on parle de style intégrés ou "embedding"
- Stocker les styles en ligne dans une balise HTML, et donc ne s'appliquant qu'au contenu géré par cette balise <xxxx>
on parle de style en ligne ou "in-lining"
- Stocker les styles entre des balises spécifiques et lorsque l'on souhaite styliser une partie très précise d'un document, ne s'étendant pas entre deux balises pré-définies classiques

Lien vers une feuille de styles

Pour réaliser un lien vers une feuille de style vous devez incorporer une instruction de ce type dans le head de votre page web.

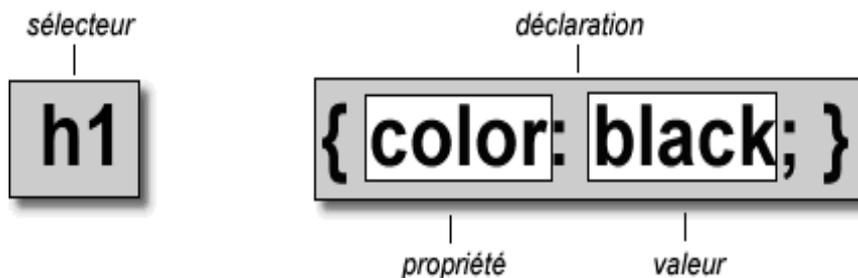
```
<head>
....
<link      href="Chemin_d'accès_au_fichier.css"      rel="stylesheet"
type="text/css" />
....
</head>
```

Les sélecteurs sur balises HTML

Une feuille de style est constituée principalement de règles. Chaque règle est, quant à elle, formée de deux parties: le sélecteur et la déclaration

Syntaxe :

Structure d'une règle CSS



Les sélecteurs simples sont des balises du HTML

Exemple :

```
h1 {
  font-family: Arial;
  color: #FF0000;
}
```

Ceci est du h1

`<h1>Ceci est du h1</h1>`

Utilisation générale :

A chaque fois que vous utiliserez la balise html h1, le texte s'écrira comme il a été défini dans la déclaration. Tous les h1 utilisés dans la page sont naturellement en police arial rouge (en plus de leurs caractéristiques par défaut)

principales balises html redéfinissables

balise	type	fonction
a	inline	lien
b	inline	gras
body	block	corps de la page
br	inline	saut de ligne forcé
div	block	bloc d'éléments
em	inline	Emphase (italique)
h1	inline	titre 1
h2	inline	titre 2
h3	inline	titre 3
h4	inline	titre 4
h5	inline	titre 5
h6	inline	titre 6
img	inline	image
li	block	élément de liste
ol	block	liste ordonnée
p	inline	paragraphe
span	inline	span
table	block	tableau
td	block	cellule de tableau
tr	block	ligne de tableau
u	inline	souligné
ul	block	liste non ordonnée

QU'EST-CE JAVASCRIPT

Présentation

Auparavant, la majorité des traitements effectués dans les documents HTML étaient réalisés par l'action du serveur et le poste client était très passif, puisque celui-ci se bornait quasiment à afficher les résultats des requêtes que lui soumettait le dit serveur ! En effet, un programme ou un script CGI (Common Gateway Interface) était exécuté par le serveur qui transmettait le résultat au poste client, celui-ci interprétait l'envoi HTML et l'affichait par le navigateur. Cette situation surchargeait énormément les serveurs et le réseau...

Aujourd'hui, via Javascript (introduit par Netscape 2.0), cette situation a changé et le navigateur est maintenant capable "d'intelligence" puisqu'il est à même de traiter des scripts embarqués coté « client », de manière indépendante au serveur (réseau non utilisé), par un traitement local.

Il est vrai que le langage JAVA est aussi capable et même plus que le Javascript, mais celui-ci est réservé à des spécialistes. Il permet beaucoup plus mais est beaucoup plus lourd, car il nécessite une bonne connaissance de la programmation (Objet), un compilateur (il est compilé à l'inverse du Javascript), un débogueur etc.

Le couple HTML-Javascript est très souple, accessible à la plupart des Webmasters et sa popularité ne cesse de grandir car il permet une réelle autonomie au poste client, même s'il ne peut accéder aux fichiers du poste client sur lequel il s'exécute !

Objectifs

Javascript est un langage de script bien adapté à la création de petits programmes simples, comme du contrôle de formulaire, de l'animation basique de page...

Javascript n'est pas adapté pour mettre en place un affichage ou de la manipulation de donnée, mais permet d'interagir avec l'utilisateur de manière beaucoup plus souple que les langages plus complets tels que JAVA



Déclenchement par événements

De par comment il est intégré aux navigateurs et au code HTML, Javascript peut programmer des réponses à des événements utilisateurs tels qu'un clic souris ou la saisie de donnée dans un champs de formulaire, faire apparaître de l'aide dans la barre d'état lorsque le pointeur de la souris passe sur une partie de la page,

Orienté Objet

Javascript intègre une forme limitée de modèle orienté objet

Les plus et ...les moins :

Plus : interprété, la phase test-essais ne nécessite pas de compilation et est donc très rapide

Plus : indépendant de la plateforme utilisée

Plus : taille réduite et vitesse d'exécution élevée

Moins : code source non dissimulable

Moins : faibles possibilités de débogage et fonctionnalités relativement limitées

Moins : nécessite une version relativement récente des navigateurs

Moins : interprété, les nuances d'interprétation d'un navigateur à l'autre, voire d'une version à une autre peuvent être considérables

Ce qu'est php

PHP (officiellement "**PHP: Hypertext Preprocessor**") est un langage de script HTML, qui fonctionne coté serveur.

Il est à noter la différence avec les autres scripts CGI écrits dans d'autres langages tels que le Perl ou le C : Au lieu d'écrire un programme avec de nombreuses lignes de commandes afin d'afficher une page HTML, vous écrivez une page HTML avec du code inclus à l'intérieur afin de réaliser une action précise

Ce qui distingue le PHP des langages de script comme le Javascript est que le code est exécuté sur le serveur. Si vous avez un script similaire sur votre serveur, le client ne reçoit que le résultat du script, sans aucun moyen d'avoir accès au code qui a produit ce résultat.

Le langage PHP possède les même fonctionnalités que les autres langages permettant d'écrire des scripts CGI, comme collecter des données, générer dynamiquement des pages web ou bien envoyer et recevoir des cookies. La plus grande qualité et le plus important avantage du langage PHP est le support d'un grand nombre de bases de données.

Les plus et ...les moins :

Plus : utilise un pré-processeur installé coté serveur, on est "sur" du fonctionnement de cet outils

Plus : indépendant de la plateforme utilisée

Plus : code source dissimulable (transparent pour l'utilisateur)

Moins : l'imbrication php html feuilles de style peut vite devenir complexe

Moins : nécessite un apprentissage plus complexe

CARACTÈRES SPÉCIAUX HTML

Table des caractères

	nbsp	 	Â	acirc	Â	Á	aacute	Á
i	iexcl	¡	Ã	atilde	Ã	á	aacute	á
¢	cent	¢	Ä	auml	Ä	â	acirc	â
£	pound	£	Å	aring	Å	ã	atilde	ã
¤	curren	¤	Æ	aelig	Æ	ä	auml	ä
¥	yen	¥	Ç	ccedil	Ç	å	aring	å
	brvbar	¦	È	egrave	È	æ	aelig	æ
§	sect	§	É	eacute	É	ç	ccedil	ç
¨	uml	¨	Ê	ecirc	Ê	è	egrave	è
©	copy	©	Ë	euml	Ë	é	eacute	é
ª	ordf	ª	Ì	igrave	Ì	ê	ecirc	ê
«	laquo	«	Í	iacute	Í	ë	euml	ë
¬	not	¬	Î	icirc	Î	ì	igrave	ì
-	shy	­	Ë	iuml	Ï	í	iacute	í
®	reg	®	Ñ	ntilde	Ñ	î	icirc	î
-	macr	¯	Ò	ograve	Ò	ï	iuml	ï
°	deg	°	Ó	oacute	Ó	ñ	ntilde	ñ
±	plusmn	±	Ô	ocirc	Ô	ò	ograve	ò
´	acute	´	Õ	otilde	Õ	ó	oacute	ó
µ	micro	µ	Ö	ouml	Ö	ô	ocirc	ô
¶	para	¶	&	times	×	õ	otilde	õ
·	middot	·	Ø	oslash	Ø	ö	ouml	ö
¸	cedil	¸	Ù	ugrave	Ù	÷	divide	÷
º	ordm	º	Ú	uacute	Ú	ø	oslash	ø
»	raquo	»	Û	ucirc	Û	ù	ugrave	ù
½	frac12	½	Ü	uuml	Ü	ú	uacute	ú
¿	iquest	¿	ß	szlig	ß	û	ucirc	û
À	agrave	À	à	agrave	à	ü	uuml	ü
						ÿ	yuml	ÿ

CONVERSION DE VIDEO QCM

exemple Quick Media Converter

récupérable sur <http://www.cocoonsoftware.com>

The screenshot shows the Quick Media Converter software interface. At the top, there are menu options: Menu, Langues, Mes réglages, and Aide. Below the menu is a toolbar with icons for Mode facile, Mode Expert, WebCam, CocoonSoftware, and CamStudio, along with a Donate button and a Quitter button. The main area is titled "Format de Conversion (Cliquez sur l'icône appropriée et Sélectionnez une option) :". Below this is a row of icons for various output formats: DivX, XviD, DVD, VCD, MP3, AAC, AAC+, PSP, Wii, and others. In the center, there is a table titled "Sélectionner des fichiers à convertir (Glisser/Déposer)". The table has columns for "Nom de la vidéo", "Taille", "Format", and "Résultat". One file named "oiseau" is selected, with a size of 4.834 mb and a format of .avi. Below the table, there are input fields for "L : 320" and "H : 240", and a "Capturer une image à : [] sec" field. At the bottom, there is a dropdown menu and two icons. Annotations with arrows point to the "oiseau" file in the table, the "Format de Conversion" section, and the output format icons.

on choisit le dossier de sortie

on charge la video à convertir... et

on demande le format désiré... !